Package 'hurricaneexposure'

February 13, 2020

Type Package

Title Explore and Map County-Level Hurricane Exposure in the United
States

Version 0.1.1

Date 2020-02-12

Description Allows users to create time series of tropical storm
exposure histories for chosen counties for a number of hazard metrics
(wind, rain, distance from the storm, etc.). This package interacts
with data available through the 'hurricaneexposuredata' package, which
is available in a 'drat' repository. To access this data package, see the
instructions at <https://github.com/geanders/hurricaneexposure>.
The size of the 'hurricaneexposuredata' package is
approximately 20 MB. This work was supported in part by grants from the National
Institute of Environmental Health Sciences (R00ES022631), the National Science
Foundation (1331399), and a NASA Applied Sciences Program/Public Health Program
Grant (NNX09AV81G).

URL <https://github.com/geanders/hurricaneexposure>

BugReports <https://github.com/geanders/hurricaneexposure/issues>

License GPL (>= 2)

LazyData TRUE

Imports data.table (>= 1.12.8), dplyr (>= 0.8.3), ggmap (>= 3.0.0),
ggplot2 (>= 3.2.1), lazyeval (>= 0.2.2), lubridate (>= 1.7.4),
mapproj (>= 1.2.6), maps (>= 3.3.0), purrr (>= 0.3.3),
RColorBrewer (>= 1.1.2), rlang (>= 0.4.2), stringr (>= 1.4.0),
tidyr (>= 1.0.0)

RoxygenNote 7.0.2

Encoding UTF-8

Depends R (>= 3.6)

Suggests hurricaneexposuredata (>= 0.1.0), knitr (>= 1.26.0), pander
(>= 0.6.3), rmarkdown (>= 2.0.0), weathermetrics (>= 1.2.2)

VignetteBuilder knitr

Additional_repositories https://geanders.github.io/drat

Language en-US

NeedsCompilation no

Author Brooke Anderson [aut, cre],
      Meilin Yan [aut],
      Joshua Ferreri [aut],
      William Crosson [ctb],
      Mohammad Al-Hamdan [ctb],
      Andrea Schumacher [ctb],
      Dirk Eddelbuettel [ctb]

Maintainer Brooke Anderson <brooke.anderson@colostate.edu>

Repository CRAN

Date/Publication 2020-02-13 14:30:02 UTC

R **topics documented:**

---

county_distance   *Hurricane exposure by distance for counties*

Description

This function takes a list of US counties,based on their 5-digit Federal Information Processing Standard (FIPS) codes, boundaries on the range of years to be considered, and thresholds for distance between each county and the storm track for the county to be considered "exposed" to the storm. Based on these inputs, the function returns a dataframe with the subset of Atlantic basin storms meeting those criteria for each of the listed counties.

Usage

```
county_distance(counties, start_year, end_year, dist_limit)
```

Arguments

counties	Character vector of the five-digit Federal Information Processing Standard (FIPS) codes for counties for which the user wants to determine storm exposure.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
dist_limit	Maximum distance, in kilometers, of how close the storm track must come to the county's population mean center to classify the county as "exposed" to the storm.

Details

For more information on how distances between counties and storm tracks are calculated for this function, see the documentation for the closest_dist dataset that comes with this package.

Value

Returns a dataframe with a row for each county-storm pair and with columns for:

- storm_id: Unique storm identifier with the storm name and year, separated by a hyphen(e.g., "Alberto-1988", "Katrina-2005")
- fips: County's 5-digit Federal Information Processing Standard (FIPS) code
- closest_date: Date (based on local time) of the closest approach of the storm to the county's population mean center.
- storm_dist: Minimum distance (in kilometers) between the storm's track and the county's population mean center.
- local_time: Local time of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.
- closest_time_utc: Time, in UTC, of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.

Note

Only counties in states in the eastern half of the United States can be processed by this function.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

  county_distance(counties = c("22071", "51700"),
              start_year = 1995, end_year = 2005,
              dist_limit = 75)

}
```

---

county_events   *Find events for storms by county*

Description

Takes a list of US counties, based on their 5-digit Federal Information Processing Standard (FIPS) codes, boundaries on the range of years to be considered, and the type of event (e.g., "flood", "tornado") and returns a list of all storms with listings for that type of event for the counties. This output is based on listings in the NOAA Storm Events database. See the help documentation for the storm_events dataset for more information and references on the data source used by this function.

Usage

```
county_events(counties, start_year, end_year, event_type)
```

Arguments

counties	Character vector of the five-digit Federal Information Processing Standard (FIPS) codes for counties for which the user wants to determine storm exposure.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
event_type	Character string with the type of event to use to identify county exposures. Options include "flood", "tornado", "wind", and "tropical_storm".

Value

Returns a dataframe with a row for each county-storm pair and with columns for:

- fips: County's 5-digit Federal Information Processing Standard (FIPS) code
- storm_id: Unique storm identifier with the storm name and year, separated by a hyphen (e.g., "Alberto-1988", "Katrina-2005")

- usa_atcf_id: United States Automated Tropical Cyclone Forecasting storm identifier
- closest_time_utc: Time, in UTC, of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.
- storm_dist: Minimum distance (in kilometers) between the storm's track and the county's population mean center.
- local_time: Local time of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.
- closest_date: Date (based on local time) of the closest approach of the storm to the county's population mean center.

Note

Of the event types that this function can pull, only "tornado" and "wind" are available for years before 1996. Therefore, this function will pull listed tornado events or wind events for all years, but for any other event type, you should not use a start year prior to 1996, as events of other types were not recorded in the database before 1996.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

county_events(counties = c("37031", "37053", "37055"),
              start_year = 1996, end_year = 2018,
              event_type = "flood")

county_events(counties = c("37055"),
              start_year = 1996, end_year = 2018,
              event_type = "tropical_storm")
}
```

---

county_rain   *Hurricane exposure by rain for counties*

Description

This function takes a list of US counties, based on their 5-digit Federal Information Processing Standard (FIPS) codes, boundaries on the range of years to be considered, thresholds for distance between each county and the storm track, as well as minimum rainfall over a certain time window for the county to be considered "exposed" to the storm. Based on these inputs, the function returns a dataframe with the subset of Atlantic basin storms meeting those criteria for each of the listed counties.

Usage

```
county_rain(
  counties,
  start_year,
  end_year,
  rain_limit,
  dist_limit,
  days_included = c(-2, -1, 0, 1)
)
```

Arguments

counties	Character vector of the five-digit Federal Information Processing Standard (FIPS) codes for counties for which the user wants to determine storm exposure.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
rain_limit	Minimum of rainfall, in millimeters, summed across the days selected to be included (days_included), that must fall in a county for the county to be classified as "exposed" to the storm.
dist_limit	Maximum distance, in kilometers, of how close the storm track must come to the county's population mean center to classify the county as "exposed" to the storm.
days_included	A numeric vector listing the days to include when calculating total precipitation. Negative numbers are days before the closest date of the storm to a county. For example, c(-1,0,1) would calculate rain for a county as the sum of the rainfall for the day before, the day of, and the day after the date when the storm center was closest to the county center. Values can range from -5 to 3 (i.e., at most, you can calculate the total rainfall from five days to three days after the day when the storm is closest to the county).

Value

Returns a dataframe with a row for each county-storm pair and with columns for:

- storm_id: Unique storm identifier with the storm name and year, separated by a hyphen(e.g., "Alberto-1988", "Katrina-2005")
- fips: County's 5-digit Federal Information Processing Standard (FIPS) code
- closest_date: Date (based on local time) of the closest approach of the storm to the county's population mean center.
- storm_dist: Minimum distance (in kilometers) between the storm's track and the county's population mean center.
- tot_precip: Cumulative rainfall, in millimeters, in the county for the days selected using the days_included option.
- local_time: Local time of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.
- closest_time_utc: Time, in UTC, of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.

References

Al-Hamdan MZ, Crosson WL, Economou SA, Estes MG, Estes SM, Hemmings SN, Kent ST, Puckette M, Quattrochi DA, Rickman DL, Wade GM, McClure LA, 2014. Environmental public health applications using remotely sensed data. Geocarto International 29(1):85-98.

North America Land Data Assimilation System (NLDAS) Daily Precipitation years 1979-2011 on CDC WONDER Online Database, released 2012. http://wonder.cdc.gov/wonder/help/Precipitation.html

Rui H, Mocko D, 2014. README Document for North America Land Data Assimilation System Phase 2 (NLDAS-2) Products. Goddard Earth Sciences Data and Information Services Center.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

county_rain(counties = c("22071", "51700"),
            start_year = 1995, end_year = 2005,
            rain_limit = 100, dist_limit = 100)
}
```

---

county_wind   *Hurricane exposure by wind for counties*

Description

This function takes a list of US counties, based on their 5-digit Federal Information Processing Standard (FIPS) codes, boundaries on the range of years to be considered, and thresholds for wind speed (in meters per second) (or, alternatively, duration of winds at or above 20 m / s in minutes) for each county to be considered "exposed" to the storm. Based on these inputs, the function returns a dataframe with the subset of Atlantic basin storms meeting those criteria for each of the listed counties.

Usage

```
county_wind(
  counties,
  start_year,
  end_year,
  wind_limit,
  wind_var = "vmax_sust",
  wind_source = "modeled"
)
```

Arguments

`counties`	Character vector of the five-digit Federal Information Processing Standard (FIPS) codes for counties for which the user wants to determine storm exposure.
`start_year`	Four-digit integer with first year to consider.
`end_year`	Four-digit integer with last year to consider.
`wind_limit`	A numeric vector of length one giving the minimum wind speed (in meters per second) or duration of winds of 20 m / s or more (in minutes) to use in the filter. The units of this variable will depend on the user's choice for the `wind_var` parameter. If the Extended Best Tracks wind radii are used as the source of the wind data, the sustained winds will only be available for cutpoints of 34 knots, 50 knots, and 64 knots, so these values should be used (e.g., to get all counties with winds of 34 knots or higher, you could use `wind_limit = 17.4`, with the limit given as a value just below 34 knots in the units meters per second).
`wind_var`	A character string giving the wind variable to use. Choices are `"vmax_sust"` (maximum sustained winds; default), `"vmax_gust"` (maximum gust winds), `"sust_dur"` (minutes of sustained winds of 20 m / s or higher) and `"gust_dur"` (minutes of gust winds of 20 m / s or higher). If the Extended Best Tracks wind radii are used as the source of wind data, the `"gust_dur"` option cannot be selected.
`wind_source`	A character string specifying the source to use for the winds. Options are `"modeled"`, for estimates based on running a wind model from Best Tracks data inputs, and `"ext_tracks"`, for estimates based on the wind radii in the Extended Best Tracks data. See the help files for the datasets `storm_winds` and `ext_tracks_wind` in the `hurricaneexposuredata` package for more details on each of these sources for wind estimates. For the gust wind estimates, these are based on applying a gust factor of 1.49 to the sustained wind estimates in both wind data sources.

Details

For more information on how wind speeds are modeled in this data, see the documentation for the `stormwindmodel` R package.

Value

Returns a dataframe with a row for each county-storm pair and with columns for:

- `storm_id`: Unique storm identifier with the storm name and year, separated by a hyphen(e.g., "Alberto-1988", "Katrina-2005")
- `fips`: County's 5-digit Federal Information Processing Standard (FIPS) code
- `max_sust`: Maximum sustained wind speed (in m / s)
- `max_gust`: Maximum gust wind speed (in m / s)
- `sust_dur`: Minutes sustained wind speed was 20 m / s or higher
- `gust_dur`: Minutes gust wind speed was 20 m / s or higher (only returned if the modeled winds are requested using `wind_source`)

- usa_atcf_id: United States Automated Tropical Cyclone Forecasting storm identifier
- closest_time_utc: Time, in UTC, of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.
- storm_dist: Minimum distance (in kilometers) between the storm's track and the county's population mean center.
- local_time: Local time of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.
- closest_date: Date (based on local time) of the closest approach of the storm to the county's population mean center.

Note

Only counties in states in the eastern half of the United States can be processed by this function. Winds are modeled at 15-minute increments, so all duration estimates (sust_dur and gust_dur in the output) will be divisible by 15.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

county_wind(counties = c("22071", "51700"),
            start_year = 1988, end_year = 2005,
            wind_limit = 20, wind_var = "vmax_sust")
}
```

---

default_map | *Create a default map with eastern US states*

Description

Creates a ggplot object with the underlying map of all states in the eastern section of the US that might be prone to hurricane-related exposure. Other lines and points can be added to the output using ggplot2 plotting functions.

Usage

```
default_map()
```

Details

Only states in the eastern half of the United States (i.e., ones prone to exposure to Atlantic basin tropical storms) are included on this map.

Value

A ggplot object that maps the states of the Eastern United States

Examples

```
default_map()
```

---

distance_exposure   *Write storm distance exposure files*

Description

This function takes an input of locations (either a vector of county FIPS or a dataframe of multi-county FIPS, with all FIPS listed for each county) and creates a dataframe with storm listings and dates that can be merged with time series of health or other outcomes, giving the dates and exposures for all storms meeting the given storm distance criteria.

Usage

```
distance_exposure(
  locations,
  start_year,
  end_year,
  dist_limit,
  out_dir,
  out_type = "csv"
)
```

Arguments

locations	Either a vector of FIPS county codes, for county-level output, or a dataframe with columns for community identifier (commun) and associated FIPS codes (fips), for multi-county community output. See the examples for the proper format for this argument.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
dist_limit	Maximum distance, in kilometers, of how close the storm track must come to the county's population mean center to classify the county as "exposed" to the storm.
out_dir	Character string giving the pathname of the directory in which to write output. This directory should already exist on your computer.
out_type	Character string giving the type of output files you'd like. Options are "csv" (default) and "rds".

Value

Writes out a directory with rain exposure files for each county or community indicated. For more on the columns in this output, see the documentation for county_rain and multi_county_rain.

Examples

```
## Not run:
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

# For these examples to work, you will need to have a directory called "tmp"
# as a subdirectory of your home directory. These examples will create new
# directories with exposure output to that "tmp" directory.

# By county
distance_exposure(locations = c("22071", "51700"),
              start_year = 1995, end_year = 2005,
              dist_limit = 75,
              out_dir = "~/tmp/storms")

# For multi-county communities
communities <- data.frame(community_name = c(rep("ny", 6), "no", "new"),
                          fips = c("36005", "36047", "36061",
                          "36085", "36081", "36119",
                          "22071", "51700"))
distance_exposure(locations = communities,
              start_year = 1995, end_year = 2005,
              dist_limit = 75,
              out_dir = "~/tmp/storms")
}

## End(Not run)
```

---

events_exposure   *Write storm events exposure files*

Description

This function takes an input of locations (either a vector of county FIPS or a dataframe of multi-county FIPS, with all FIPS listed for each county) and creates a dataframe with storm listings and dates that can be merged with time series of health or other outcomes, giving the dates and exposures for all storms meeting the given storm events criteria.

Usage

```
events_exposure(
  locations,
```

```
    start_year,
    end_year,
    event_type,
    out_dir,
    out_type = "csv"
)
```

Arguments

locations	Either a vector of FIPS county codes, for county-level output, or a dataframe with columns for community identifier (commun) and associated FIPS codes (fips), for multi-county community output. See the examples for the proper format for this argument.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
event_type	Character string with the type of event to use to identify county exposures. Options include "flood", "tornado", "wind", and "tropical_storm".
out_dir	Character string with the filepath to the directory where the data will be saved
out_type	Character string with the type of file to save to. Options are "csv" for a comma-separated file (default) and "rds" for an R object file.

Value

Writes out a directory with rain exposure files for each county or community indicated. For more on the columns in this output, see the documentation for county_rain and multi_county_rain.

Examples

```
## Not run:
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

# To run this example, you will need to have a directory named "tmp"
# as a subdirectory of your home directory.

# By county
events_exposure(locations = c("22071", "51700"),
                start_year = 1995, end_year = 2005,
                event_type = "flood",
                out_dir = "~/tmp/storms")
                }

## End(Not run)
```

---

filter_storm_data *Filter hurricane datasets*

Description

This function is a helper function for many of the the other functions in this package that measure exposure.

Usage

```
filter_storm_data(
  counties = NULL,
  storm = NULL,
  year_range = NULL,
  distance_limit = NULL,
  rain_limit = NULL,
  include_rain = FALSE,
  days_included = NULL,
  output_vars = c("fips")
)
```

Arguments

counties	A character vector listing all 5-digit county FIPS codes for a subset of counties.
storm	A character string giving the storm ID (e.g., "Floyd-1999")
year_range	A numeric vector of length two with the starting and ending year to subset to.
distance_limit	A numeric vector of length one giving the maximum distance (in kilometers) to use in the filter
rain_limit	A numeric vector of length one giving the minimum rain (in millimeters) to use in the filter
include_rain	A logical specifying whether to pull in rain data to use in the filter or give as output (default is FALSE)
days_included	A numeric vector giving the lag numbers for days to include when calculating the total rain over the storm period (e.g., c(-1,0,1,2) would calculate the rain from the day before the storm until two days after the storm). Values in this vector cannot be lower than -3 or higher than 3.
output_vars	A character vector listing all the columns to include in the output.

Value

A dataframe with storms filtered based on the input criteria to the function. Columns in the output will vary depending on the user's selections for the output_vars argument.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

filter_storm_data(counties = c("22071", "51700"), year_range = c(1988, 2011),
                  distance_limit = 250, rain_limit = 150,
                  include_rain = TRUE, days_included = c(-1, 0, 1),
                  output_vars = c("fips", "storm_id", "closest_date",
                                      "storm_dist", "tot_precip"))
filter_storm_data(storm = "Floyd-1999", include_rain = TRUE,
                  days_included = c(-1, 0, 1),
                  output_vars = c("fips", "tot_precip"))
}
```

---

filter_wind_data	*Filter hurricane wind dataset*

Description

This function is a helper function for many of the the other functions in this package that measure wind exposure.

Usage

```
filter_wind_data(
  counties = NULL,
  storm = NULL,
  year_range = NULL,
  wind_limit = NULL,
  output_vars = "fips",
  wind_var = "vmax_sust",
  wind_source = "modeled"
)
```

Arguments

counties	A character vector listing all 5-digit county FIPS codes for a subset of counties.
storm	A character string giving the storm ID (e.g., "Floyd-1999")
year_range	A numeric vector of length two with the starting and ending year to subset to.
wind_limit	A numeric vector of length one giving the minimum wind speed (in meters per second) or duration of winds of 20 m / s or more (in minutes) to use in the filter. The units of this variable will depend on the user's choice for the wind_var parameter. If the Extended Best Tracks wind radii are used as the source of the wind data, the sustained winds will only be available for cutpoints of 34 knots, 50 knots, and 64 knots, so these values should be used (e.g., to get all counties

with winds of 34 knots or higher, you could use `wind_limit = 17.4`, with the limit given as a value just below 34 knots in the units meters per second).

output_vars   A character vector listing all the columns to include in the output.

wind_var   A character string giving the wind variable to use. Choices are `"vmax_sust"` (maximum sustained winds; default), `"vmax_gust"` (maximum gust winds), `"sust_dur"` (minutes of sustained winds of 20 m / s or higher) and `"gust_dur"` (minutes of gust winds of 20 m / s or higher). If the Extended Best Tracks wind radii are used as the source of wind data, the `"gust_dur"` option cannot be selected.

wind_source   A character string specifying the source to use for the winds. Options are `"modeled"`, for estimates based on running a wind model from Best Tracks data inputs, and `"ext_tracks"`, for estimates based on the wind radii in the Extended Best Tracks data. See the help files for the datasets `storm_winds` and `ext_tracks_wind` in the `hurricaneexposuredata` package for more details on each of these sources for wind estimates. For the gust wind estimates, these are based on applying a gust factor of 1.49 to the sustained wind estimates in both wind data sources.

Value

A dataframe with storms filtered based on the input criteria to the function. Columns in the output will vary depending on the user's selections for the `output_vars` argument.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

filter_wind_data(counties = c("22071", "51700"), year_range = c(1988, 2011),
                 wind_limit = 20,
                 output_vars = c("fips", "storm_id", "vmax_sust"))
}
```

---

get_eastern_map   *Get map data for eastern US states*

Description

Get map data for eastern US states

Usage

```
get_eastern_map(map = "county")
```

Arguments

map   A character string giving the map database from which to pull

Value

A dataframe with map data pulled using the `map_data` function in `ggplot2`, filtered to states in the eastern half of the United States.

---

hurr_choropleth   *Create a map customized for this package*

Description

Creates a county choropleth map customized for displaying hurricane exposure. This function is used as a helper function within other mapping functions in the package.

Usage

```
hurr_choropleth(
  map_data,
  metric = "distance",
  wind_var = "vmax_sust",
  wind_source = "modeled"
)
```

Arguments

map_data   A dataframe with columns with FIPS numbers (in numeric class) for all counties
                 in the eastern US (`region`) and the exposure value (`value`)

metric   Character string giving the metric to plot. Current options are `"distance"`,
                 `"wind"`, and `"rainfall"`. These options are used to customize the color palette
                 and scale of the choropleth map produced by this function.

wind_var   A character string giving the wind variable to use. Choices are `"vmax_sust"`
                 (maximum sustained winds; default), `"vmax_gust"` (maximum gust winds),
                 `"sust_dur"` (minutes of sustained winds of 20 m / s or higher) and `"gust_dur"`
                 (minutes of gust winds of 20 m / s or higher). If the Extended Best Tracks
                 wind radii are used as the source of wind data, the `"gust_dur"` option cannot
                 be selected.

wind_source   A character string specifying the source to use for the winds. Options are
                 `"modeled"`, for estimates based on running a wind model from Best Tracks
                 data inputs, and `"ext_tracks"`, for estimates based on the wind radii in the
                 Extended Best Tracks data. See the help files for the datasets `storm_winds` and
                 `ext_tracks_wind` in the `hurricaneexposuredata` package for more details on
                 each of these sources for wind estimates. For the gust wind estimates, these are
                 based on applying a gust factor of 1.49 to the sustained wind estimates in both
                 wind data sources.

Details

The function only maps counties in states likely to be exposed to Atlantic basin tropical storms.

Value

A `ggplot` object with a map of hurricane exposure in eastern US counties

---

| `interp_track` | *Interpolate a storm track* |

Description

This function takes a wider-spaced storm track (e.g., every 6 hours) and interpolates to every 15 minutes. To do this, it uses natural cubic spline interpolation using the 'spline' function from the 'stats' package. The track is only interpolated if there are three or more observations on the central location of the storm (this is almost always the case for storms tracked in the HURDAT2 dataset).

Usage

```
interp_track(track)
```

Arguments

track   A dataframe with hurricane track data for a single storm

Value

A dataframe with hurricane track data for a single storm, interpolated to 15-minute intervals.

| `map_counties` | *Map counties* |

Description

Map counties

Usage

```
map_counties(
  storm,
  metric = "distance",
  wind_var = "vmax_sust",
  days_included = c(-2, -1, 0, 1),
  add_track = TRUE,
  wind_source = "modeled"
)
```

Arguments

storm	Character string giving the name of the storm to plot (e.g., "Floyd-1999")
metric	Character string giving the metric to plot. Current options are `"distance"`, `"wind"`, and `"rainfall"`. These options are used to customize the color palette and scale of the choropleth map produced by this function.
wind_var	A character string giving the wind variable to use. Choices are `"vmax_sust"` (maximum sustained winds; default), `"vmax_gust"` (maximum gust winds), `"sust_dur"` (minutes of sustained winds of 20 m / s or higher) and `"gust_dur"` (minutes of gust winds of 20 m / s or higher). If the Extended Best Tracks wind radii are used as the source of wind data, the `"gust_dur"` option cannot be selected.
days_included	A numeric vector listing the days to include when calculating total precipitation. Negative numbers are days before the closest date of the storm to a county. For example, `c(-1,0,1)` would calculate rain for a county as the sum of the rainfall for the day before, the day of, and the day after the date when the storm center was closest to the county center. Values can range from -5 to 3 (i.e., at most, you can calculate the total rainfall from five days to three days after the day when the storm is closest to the county).
add_track	TRUE / FALSE of whether to add the storm's track to the map. The default is TRUE.
wind_source	A character string specifying the source to use for the winds. Options are `"modeled"`, for estimates based on running a wind model from Best Tracks data inputs, and `"ext_tracks"`, for estimates based on the wind radii in the Extended Best Tracks data. See the help files for the datasets `storm_winds` and `ext_tracks_wind` in the `hurricaneexposuredata` package for more details on each of these sources for wind estimates. For the gust wind estimates, these are based on applying a gust factor of 1.49 to the sustained wind estimates in both wind data sources.

Value

This function creates a choropleth map of counties in the eastern part of the United States, showing distance from a storm track, maximum wind speed (or duration of winds at or above 20 m / s), or total rainfall over a given window of one or more days near the date of the storm's closest approach.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

map_counties("Floyd-1999", metric = "rainfall", days_included = c(-2, -1, 0, 1))

map_counties("Katrina-2005", metric = "wind")
map_counties("Katrina-2005", metric = "wind", wind_var = "vmax_gust")
map_counties("Katrina-2005", metric = "wind", wind_var = "sust_dur")
map_counties("Katrina-2005", metric = "wind", wind_source = "ext_tracks")
```

```
#' map_counties("Michael-2018", metric = "wind")
map_counties("Michael-2018", metric = "wind", wind_var = "vmax_gust")
map_counties("Michael-2018", metric = "wind", wind_source = "ext_tracks")
}
```

map_distance_exposure   *Map counties with distance exposure*

Description

Map counties as "exposed" or "unexposed" based on the criterion that the storm came within a given distance (specified by dist_limit) of the county's population mean center.

Usage

```
map_distance_exposure(storm, dist_limit, add_track = TRUE)
```

Arguments

storm	Character string giving the name of the storm to plot (e.g., "Floyd-1999")
dist_limit	Maximum distance, in kilometers, of how close the storm track must come to the county's population mean center to classify the county as "exposed" to the storm.
add_track	TRUE / FALSE of whether to add the storm's track to the map. The default is TRUE.

Value

Plots a map showing whether eastern US counties were exposed or unexposed to a specific storm based on a distance criterion.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {


floyd_map <- map_distance_exposure(storm = "Floyd-1999", dist_limit = 75)
floyd_map

allison_map <- map_distance_exposure(storm = "Allison-2001",
                                     dist_limit = 75)
map_tracks("Allison-2001", plot_points = FALSE, plot_object = allison_map)
}
```

---

map_event_exposure   *Map county-level exposure based on reported events*

Description

Map counties as "exposed" or "unexposed" based on the criterion that the county had an event listing of a specified type in the NOAA Storm Events database. For more information on the underlying data, see the helpfile for the storm_events dataset.

Usage

```
map_event_exposure(storm_id, event_type, add_track = TRUE)
```

Arguments

storm_id   Character vector with the storm for which to map events (e.g., "Katrina-2005")

event_type   Character string with the type of event to use to identify county exposures. Options include "flood", "tornado", "wind", and "tropical_storm".

add_track   TRUE / FALSE of whether to add the storm's track to the map. The default is TRUE.

Value

A map showing whether eastern US counties were exposed or unexposed to a specific storm based on event listings.

Note

Note that flood events are not available for any year before 1996.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

map_event_exposure(storm_id = "Floyd-1999", event_type = "flood")
map_event_exposure(storm_id = "Floyd-1999", event_type = "tornado")
map_event_exposure(storm_id = "Floyd-1999", event_type = "wind")
map_event_exposure(storm_id = "Floyd-1999", event_type = "tropical_storm")

map_event_exposure(storm_id = "Florence-2018", event_type = "flood")
map_event_exposure(storm_id = "Florence-2018", event_type = "tropical_storm")

map_event_exposure(storm_id = "Michael-2018", event_type = "wind")
map_event_exposure(storm_id = "Michael-2018", event_type = "tropical_storm")
}
```

---

map_rain_exposure *Map counties with rain exposure*

Description

Map counties as "exposed" or "unexposed" based on the criteria that the storm came within a given distance (specified by dist_limit) of the county's population mean center and a certain amount of rain rain_limit fell during a specified window of days (days_included).

Usage

```
map_rain_exposure(
  storm,
  rain_limit,
  dist_limit,
  days_included = c(-2, -1, 0, 1),
  add_track = TRUE
)
```

Arguments

storm	Character string giving the name of the storm to plot (e.g., "Floyd-1999")
rain_limit	Minimum of rainfall, in millimeters, summed across the days selected to be included (days_included), that must fall in a county for the county to be classified as "exposed" to the storm.
dist_limit	Maximum distance, in kilometers, of how close the storm track must come to the county's population mean center to classify the county as "exposed" to the storm.
days_included	A numeric vector listing the days to include when calculating total precipitation. Negative numbers are days before the closest date of the storm to a county. For example, c(-1,0,1) would calculate rain for a county as the sum of the rainfall for the day before, the day of, and the day after the date when the storm center was closest to the county center. Values can range from -5 to 3 (i.e., at most, you can calculate the total rainfall from five days to three days after the day when the storm is closest to the county).
add_track	TRUE / FALSE of whether to add the storm's track to the map. The default is TRUE.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

map_rain_exposure(storm = "Floyd-1999", rain_limit = 50, dist_limit = 100)
```

```
# Example of customizine track appearance
allison_map <- map_rain_exposure(storm = "Allison-2001", rain_limit = 125,
                                 dist_limit = 100, days_included = -5:3,
                                 add_track = FALSE)
map_tracks("Allison-2001", plot_object = allison_map, plot_points = TRUE)
}
```

---

map_tracks   *Plot Atlantic basin hurricane tracks*

Description

Plot the tracks of any selected storms in the hurricane tracking dataset for the Atlantic basin. This
function allows you to plot a new map or add the tracks to an existing ggplot object.

Usage

```
map_tracks(
  storms,
  plot_object = NULL,
  padding = 2,
  plot_points = FALSE,
  alpha = 1,
  color = "firebrick"
)
```

Arguments

storms   Character vector with the names of all storms to plot. This parameter must use
                the unique storm identifiers from the 'storm_id' column of the 'hurr_tracks'
                dataframe.

plot_object   NULL or the name of a ggplot object to use as the underlying plot object. If
                NULL, the function will generate a new map of the eastern US states using
                'default_map'.

padding   Numerical value giving the number of degrees to add to the outer limits of the
                plot object (or default map if 'plot_object' is left as NULL) when cropping hur-
                ricane tracks.

plot_points   TRUE / FALSE indicator of whether to include points, as well as lines, when
                plotting the hurricane tracks. These points show the times for which observa-
                tions were recorded in the hurricane track data.

alpha   Numerical value designating the amount of transparency to use for plotting
                tracks.

color   Character string giving the color to use to plot the tracks.

Value

Returns a ggplot object with plotting data for the storm tracks of the selected storms. This object can be printed directly or added to with other `ggplot2` commands.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

map_tracks(storms = "Sandy-2012")
map_tracks(storms = "Floyd-1999", plot_points = TRUE)
map_tracks(storms = c("Sandy-2012", "Floyd-1999"))
a <- map_tracks(storms = "Sandy-2012", color = "blue", alpha = 0.3)
b <- map_tracks(storms = "Floyd-1999", plot_object = a)
b
}
```

---

map_wind_exposure   *Map counties with wind exposure*

Description

Map counties as "exposed" or "unexposed" based on the criterion that the storm had a certain wind speed (specified by `wind_limit`, in meters per second).

Usage

```
map_wind_exposure(
  storm,
  wind_var = "vmax_sust",
  wind_limit,
  add_track = TRUE,
  wind_source = "modeled"
)
```

Arguments

storm   A character string giving the storm ID (e.g., "Floyd-1999")

wind_var   A character string giving the wind variable to use. Choices are `"vmax_sust"` (maximum sustained winds; default), `"vmax_gust"` (maximum gust winds), `"sust_dur"` (minutes of sustained winds of 20 m / s or higher) and `"gust_dur"` (minutes of gust winds of 20 m / s or higher). If the Extended Best Tracks wind radii are used as the source of wind data, the `"gust_dur"` option cannot be selected.

wind_limit   A numeric vector of length one giving the minimum wind speed (in meters per
              second) or duration of winds of 20 m / s or more (in minutes) to use in the filter.
              The units of this variable will depend on the user's choice for the wind_var
              parameter. If the Extended Best Tracks wind radii are used as the source of the
              wind data, the sustained winds will only be available for cutpoints of 34 knots,
              50 knots, and 64 knots, so these values should be used (e.g., to get all counties
              with winds of 34 knots or higher, you could use wind_limit = 17.4, with the
              limit given as a value just below 34 knots in the units meters per second).

add_track   TRUE / FALSE of whether to add the storm's track to the map. The default is
              TRUE.

wind_source   A character string specifying the source to use for the winds. Options are
              "modeled", for estimates based on running a wind model from Best Tracks
              data inputs, and "ext_tracks", for estimates based on the wind radii in the
              Extended Best Tracks data. See the help files for the datasets storm_winds and
              ext_tracks_wind in the hurricaneexposuredata package for more details on
              each of these sources for wind estimates. For the gust wind estimates, these are
              based on applying a gust factor of 1.49 to the sustained wind estimates in both
              wind data sources.

Value

Plots a map showing whether eastern US counties were exposed or unexposed to a specific storm
based on a wind criterion.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

map_wind_exposure(storm = "Beryl-1988", wind_limit = 15)
}
```

---

multi_county_distance   *Hurricane exposure by distance for communities*

Description

This function takes a dataframe with multi-county communities and returns a community-level
dataframe of "exposed" storms, based on the shortest distance between the storm's track and the
population-based centers of each county in the community.

Usage

```
multi_county_distance(communities, start_year, end_year, dist_limit)
```

Arguments

communities   A dataframe with the FIPS codes for all counties within each community. It must
              include columns with a column identifier (commun) and with the FIPS codes of
              counties included in each community (fips). See the example code.

start_year   Four-digit integer with first year to consider.

end_year   Four-digit integer with last year to consider.

dist_limit   Maximum distance, in kilometers, of how close the storm track must come to
              the county's population mean center to classify the county as "exposed" to the
              storm.

Value

Returns the same type dataframe as county_distance, but with storms listed by community instead
of county.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

communities <- data.frame(community_name = c(rep("ny", 6), "no", "new"),
                          fips = c("36005", "36047", "36061",
                                   "36085", "36081", "36119",
                                   "22071", "51700"))
distance_df <- multi_county_distance(communities = communities,
                                     start_year = 1995, end_year = 2005,
                                     dist_limit = 75)
}
```

---

multi_county_events   *Hurricane exposure by events for communities*

Description

This function takes a dataframe with multi-county communities and returns a community-level
dataframe of "exposed" storms, based on the type of the event.

Usage

```
multi_county_events(communities, start_year, end_year, event_type)
```

Arguments

communities	A dataframe with the FIPS codes for all counties within each community. It must include columns with a column identifier (commun) and with the FIPS codes of counties included in each community (fips). See the example code.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
event_type	Character string with the type of event to use to identify county exposures. Options include "flood", "tornado", "wind", and "tropical_storm".

Value

Returns the same type dataframe as county_events, but with storms listed by community instead of county.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

communities <- data.frame(community_name = c(rep("ny", 6), "no", "new"),
                          fips = c("36005", "36047", "36061",
                                   "36085", "36081", "36119",
                                   "22071", "51700"))
distance_df <- multi_county_events(communities = communities,
                                   start_year = 1995, end_year = 2005,
                                   event_type = "flood")
}
```

---

multi_county_rain   *Hurricane exposure by rain for communities*

Description

This function takes a dataframe with multi-county communities (see example for the proper format) and returns a community-level dataframe of storms to which the community was exposed, based on the average distance between the storm's track and the population-based centers of each county in the community and the given threshold of rainfall, summed over the days included in the rainfall measurement.

Usage

```
multi_county_rain(
  communities,
  start_year,
  end_year,
```

```
    rain_limit,
    dist_limit,
    days_included = c(-2, -1, 0, 1)
)
```

Arguments

communities	A dataframe with the FIPS codes for all counties within each community. It must include columns with a column identifier (commun) and with the FIPS codes of counties included in each community (fips). See the example code.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
rain_limit	Minimum of rainfall, in millimeters, summed across the days selected to be included (days_included), that must fall in a county for the county to be classified as "exposed" to the storm.
dist_limit	Maximum distance, in kilometers, of how close the storm track must come to the county's population mean center to classify the county as "exposed" to the storm.
days_included	A numeric vector listing the days to include when calculating total precipitation. Negative numbers are days before the closest date of the storm to a county. For example, c(-1,0,1) would calculate rain for a county as the sum of the rainfall for the day before, the day of, and the day after the date when the storm center was closest to the county center. Values can range from -5 to 3 (i.e., at most, you can calculate the total rainfall from five days to three days after the day when the storm is closest to the county).

Value

Returns a dataframe with a row for each county-storm pair and with columns for:

- commun: Each community's unique id
- storm_id: Unique storm identifier with the storm name and year, separated by a hyphen(e.g., "Alberto-1988", "Katrina-2005")
- closest_date: Date (based on local time) of the closest approach of the storm to the county's population mean center.
- local_time: Local time of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.
- closest_time_utc: Time, in UTC, of the closest approach of the storm to the county's population mean center, based on storm tracks linearly interpolated to 15-minute increments.
- mean_dist: Average of the minimum distance (in kilometers) between the storm's track and the population mean centers of all the counties in the community.
- mean_rain: Average of cumulative rainfall, in millimeters, in the counties in the community for the days selected using the days_included option.
- min_dist: The smallest minimum distance (in kilometers) between the storm's track and the population mean centers of any of the counties in the community.
- max_rain: The maximum cumulative rainfall, in millimeters, in any of the counties in the community for the days selected using the days_included option.

References

Al-Hamdan MZ, Crosson WL, Economou SA, Estes MG, Estes SM, Hemmings SN, Kent ST, Puckette M, Quattrochi DA, Rickman DL, Wade GM, McClure LA, 2014. Environmental public health applications using remotely sensed data. Geocarto International 29(1):85-98.

North America Land Data Assimilation System (NLDAS) Daily Precipitation years 1979-2011 on CDC WONDER Online Database, released 2012. http://wonder.cdc.gov/wonder/help/Precipitation.html

Rui H, Mocko D, 2014. README Document for North America Land Data Assimilation System Phase 2 (NLDAS-2) Products. Goddard Earth Sciences Data and Information Services Center.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

communities <- data.frame(community_name = c(rep("ny", 6), "no", "new"),
                          fips = c("36005", "36047", "36061",
                                   "36085", "36081", "36119",
                                   "22071", "51700"))
rain_storm_df <- multi_county_rain(communities = communities,
                                   start_year = 1995, end_year = 2005,
                                   rain_limit = 100, dist_limit = 100)
}
```

---

multi_county_wind   *Hurricane exposure by wind for communities*

Description

This function takes a dataframe with multi-county communities and returns a community-level dataframe of "exposed" storms, based on the highest of the maximum sustained wind speed for each county in the community.

Usage

```
multi_county_wind(communities, start_year, end_year, wind_limit)
```

Arguments

communities	A dataframe with the FIPS codes for all counties within each community. It must include columns with a column identifier (commun) and with the FIPS codes of counties included in each community (fips). See the example code.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.

wind_limit   A numeric vector of length one giving the minimum wind speed (in meters per second) or duration of winds of 20 m / s or more (in minutes) to use in the filter. The units of this variable will depend on the user's choice for the `wind_var` parameter. If the Extended Best Tracks wind radii are used as the source of the wind data, the sustained winds will only be available for cutpoints of 34 knots, 50 knots, and 64 knots, so these values should be used (e.g., to get all counties with winds of 34 knots or higher, you could use `wind_limit = 17.4`, with the limit given as a value just below 34 knots in the units meters per second).

Value

Returns the same type dataframe as `county_rain`, but with storms listed by community instead of county.

Note

This function currently will only input a threshold for the sustained wind metric. If you would like to use gust winds or duration of winds, you will need to use the `county_wind` function to pull storms and aggregate to the multi-county community level yourself.

Examples

```
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

communities <- data.frame(community_name = c(rep("ny", 6), "no", "new"),
                          fips = c("36005", "36047", "36061",
                                   "36085", "36081", "36119",
                                   "22071", "51700"))
wind_df <- multi_county_wind(communities = communities,
                             start_year = 1988, end_year = 2005,
                             wind_limit = 20)
}
```

---

rain_exposure   *Write storm rain exposure files*

Description

This function takes an input of locations (either a vector of county FIPS or a dataframe of multi-county FIPS, with all FIPS listed for each county; see examples) and creates time series dataframes with the dates and exposures for all storms meeting the given rainfall and storm distance criteria. These exposure time series can then be merged with other time series (e.g., community-specific daily counts of health outcomes).

Usage

```
rain_exposure(
  locations,
  start_year,
  end_year,
  rain_limit,
  dist_limit,
  days_included = c(-2, -1, 0, 1),
  out_dir,
  out_type = "csv"
)
```

Arguments

locations	Either a vector of FIPS county codes, for county-level output, or a dataframe with columns for community identifier (commun) and associated FIPS codes (fips), for multi-county community output. See the examples for the proper format for this argument.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
rain_limit	Minimum of rainfall, in millimeters, summed across the days selected to be included (days_included), that must fall in a county for the county to be classified as "exposed" to the storm.
dist_limit	Maximum distance, in kilometers, of how close the storm track must come to the county's population mean center to classify the county as "exposed" to the storm.
days_included	A numeric vector listing the days to include when calculating total precipitation. Negative numbers are days before the closest date of the storm to a county. For example, c(-1,0,1) would calculate rain for a county as the sum of the rainfall for the day before, the day of, and the day after the date when the storm center was closest to the county center. Values can range from -5 to 3 (i.e., at most, you can calculate the total rainfall from five days to three days after the day when the storm is closest to the county).
out_dir	Character string giving the pathname of the directory in which to write output. This directory should already exist on your computer.
out_type	Character string giving the type of output files you'd like. Options are "csv" (default) and "rds".

Value

This function writes out rain exposure files for each county or community indicated to the specified output directory (out_dir). For more details on the columns in the output files, see the documentation for county_rain and multi_county_rain.

Examples

```
## Not run:
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace("hurricaneexposuredata", quietly = TRUE)) {

# For these examples, you need to have a directory in your home
# directory called "tmp".

# By county
rain_exposure(locations = c("22071", "51700"),
              start_year = 1995, end_year = 2005,
              rain_limit = 100, dist_limit = 100,
              out_dir = "~/tmp/storms")

# For multi-county communities
communities <- data.frame(community_name = c(rep("ny", 6), "no", "new"),
                          fips = c("36005", "36047", "36061",
                          "36085", "36081", "36119",
                          "22071", "51700"))
rain_exposure(locations = communities,
              start_year = 1995, end_year = 2005,
              rain_limit = 100, dist_limit = 100,
              out_dir = "~/tmp/storms")
}

## End(Not run)
```

---

wind_exposure	*Write storm wind exposure files*

Description

This function takes an input of locations (either a vector of county FIPS or a dataframe of multi-county FIPS, with all FIPS listed for each county) and creates time series dataframes that can be merged with health time series, giving the dates and exposures for all storms meeting the given storm wind criteria.

Usage

```
wind_exposure(
  locations,
  start_year,
  end_year,
  wind_limit,
  wind_var,
  out_dir,
  out_type = "csv"
)
```

Arguments

locations	Either a vector of FIPS county codes, for county-level output, or a dataframe with columns for community identifier (commun) and associated FIPS codes (fips), for multi-county community output. See the examples for the proper format for this argument.
start_year	Four-digit integer with first year to consider.
end_year	Four-digit integer with last year to consider.
wind_limit	Vector giving the wind speed (in m / s) to use as a threshold for classifying a county as "exposed" to a specific storm.
wind_var	A character string giving the wind variable to use. Choices are ″vmax_sust″ (maximum sustained winds; default), ″vmax_gust″ (maximum gust winds), ″sust_dur″ (minutes of sustained winds of 20 m / s or higher) and ″gust_dur″ (minutes of gust winds of 20 m / s or higher). If the Extended Best Tracks wind radii are used as the source of wind data, the ″gust_dur″ option cannot be selected.
out_dir	Character string giving the pathname of the directory in which to write output. This directory should already exist on your computer.
out_type	Character string giving the type of output files you'd like. Options are ″csv″ (default) and ″rds″.

Value

Writes out a directory with rain exposure files for each county or community indicated. For more on the columns in this output, see the documentation for [county_wind](#) and [multi_county_wind](#).

Note

This function allows you to use different wind variables (sustained winds, which is the default; gust winds; duration of sustained winds; duration of gust winds) when pulling exposures by county. However, if pulling multi-county communities, currently only the sustained winds metric can be used with this function.

Examples

```
## Not run:
# Ensure that data package is available before running the example.
#  If it is not, see the `hurricaneexposure` package vignette for details
# on installing the required data package.
if (requireNamespace(″hurricaneexposuredata″, quietly = TRUE)) {

# You will need a directory named ″tmp″ in your home directory to
# run these examples.

# By county
wind_exposure(locations = c(″22071″, ″51700″),
              start_year = 1988, end_year = 2005,
              wind_limit = 10,
              out_dir = ″~/tmp/storms″)
```

```
# For multi-county communities
communities <- data.frame(community_name = c(rep("ny", 6), "no", "new"),
                          fips = c("36005", "36047", "36061",
                          "36085", "36081", "36119",
                          "22071", "51700"))
wind_exposure(locations = communities,
              start_year = 1988, end_year = 2005,
              wind_limit = 10,
              out_dir = "~/tmp/storms")
}

## End(Not run)
```

Index