

Using the `hurricaneexposure` package

Brooke Anderson

2020-02-12

Package overview

This package allows users to explore and map data of county-level exposures to Atlantic-basin tropical storms between 1988 and 2018 for a number of storm hazards (e.g., wind, rain, flood events, distance from the storm track) for counties in the eastern half of the United States. Users can map exposures by county for a selected storm and can also identify all storms to which selected counties were exposed, based on user-specified thresholds (for example, the package allows the user to identify all storms that brought wind of 34 knots or higher to Miami-Dade County in Florida).

If you use this package and the data in the associated data package (`hurricaneexposedata`) for research, please cite both packages. In particular, be sure to include the version of the packages that you used, as this will make your research project more reproducible, and the data will likely be updated as we get further years of data and find improved ways to measure tropical storm exposure. The two packages can be cited as:

- Anderson B, Yan M, Ferreri J, Crosson W, Al-Hamdan M, Schumacher A and Eddelbuettel D (2020). *hurricaneexposure: Explore and Map County-Level Hurricane Exposure in the United States*. R package version 0.1.1, <URL: <https://github.com/geanders/hurricaneexposure>>.
- Anderson B, Schumacher A, Crosson W, Al-Hamdan M, Yan M, Ferreri J, Chen Z, Quiring S and Guikema S (2020). *hurricaneexposedata: Data Characterizing Exposure to Hurricanes in United States Counties*. R package version 0.1.0, <URL: <https://github.com/geanders/hurricaneexposedata>>.

To generate BibTex entries for the packages, you can use the `citation` function in R.

Required set-up to use this package

This package depends on data in a data package (`hurricaneexposedata`) that is available through a drat repository on GitHub. To use the `hurricaneexposure` package, you will need to install `hurricaneexposedata` on your computer. You can do that by adding the drat archive to the list of repositories your system will query when adding and updating R packages. Once you do this, you can install the `hurricaneexposedata` using the `install.packages` function (and later update it using the `update.packages` function):

```
library(drat)
addRepo("geanders")
install.packages("hurricaneexposedata")
```

You will want to have the latest version of the `hurricaneexposedata` package. If you have previously installed `hurricaneexposedata`, you may need to re-run the above code if you update the `hurricaneexposure` package, to update to the latest data.

The `hurricaneexposedata` data package includes data that characterizes county-level exposure to tropical storms in counties in the eastern half of the United States between 1988 and 2015 (for some hazards, exposure data is only included for a subset of these years). Tropical storms that did not pass within at least 250 km of at least one US county were excluded from these datasets. The following datasets are included with the `hurricaneexposedata` data package:

- `county_centers`: Location of United States county centers of population

- `hurr_tracks`: Storm tracks for Atlantic-basin storms, 1988-2018
- `closest_dist`: Closest distances between counties and a storm track, for Atlantic-basin storms, 1988-2018
- `rain`: Rainfall for US counties during Atlantic basin tropical storms, 1988-2011; daily rainfall is given from five days before to three days after the storm's closest approach to the county
- `storm_winds`: Modeled county wind speeds for Atlantic-basin storms, 1988-2018
- `storm_events`: Listings from the [NOAA Storm Events database](#) that occurred near in time and location to tropical storms, 1988-2018. This database changed the types of events it reported in 1996, which should be considered when using the data.
- `ext_tracks_wind`: Estimated county wind speeds for Atlantic-basin storms, 1988-2018, based on the wind radii listed in the [HURDAT2](#)

Once you've installed and loaded `hurricaneexposedata`, you can load the included data using the `data` function. For example, you can access the data on hurricane tracks (`hurr_tracks`) using:

```
library(hurricaneexposedata)
data("hurr_tracks")
head(hurr_tracks)
```

```
## # A tibble: 6 x 6
##   storm_id    usa_atcf_id date      latitude longitude  wind
##   <chr>      <chr>    <chr>      <dbl>    <dbl> <dbl>
## 1 Alberto-1988 AL011988 198808051800 32      -77.5 20
## 2 Alberto-1988 AL011988 198808060000 32.8    -76.2 20
## 3 Alberto-1988 AL011988 198808060600 34      -75.2 20
## 4 Alberto-1988 AL011988 198808061200 35.2    -74.6 25
## 5 Alberto-1988 AL011988 198808061800 37      -73.5 25
## 6 Alberto-1988 AL011988 198808070000 38.7    -72.4 25
```

For each dataset included in `hurricaneexposedata`, you can see the helpfiles for the data for more information (e.g., `?hurr_tracks` to read documentation on the hurricane tracks data shown above). This data documentation includes both explanations of how the dataset was created, definitions of the variables included in each dataset, and the sources of data included.

The following table shows the storms covered by the `hurricaneexposedata` data package, as well as which hazard metrics are available for each year. All storms passed within 250 km of at least one U.S. county.

Year	Storms	Distance	Wind	Rain	Flood	Tornado
1988	Alberto, Beryl, Chris, Florence, Gilbert, Keith, AL13, AL14, AL17	x	x	x		x
1989	Allison, Chantal, Hugo, Jerry	x	x	x		x
1990	AL01, Bertha, Marco	x	x	x		x
1991	Ana, Bob, Fabian, AL12	x	x	x		x
1992	AL02, Andrew, Danielle, Earl	x	x	x		x
1993	AL01, Arlene, Emily	x	x	x		x
1994	Alberto, AL02, Beryl, Gordon	x	x	x		x
1995	Allison, Dean, Erin, Gabrielle, Jerry, Opal	x	x	x		x
1996	Arthur, Bertha, Edouard, Fran, Josephine	x	x	x	x	x
1997	AL01, Ana, Danny	x	x	x	x	x

Year	Storms	Distance	Wind	Rain	Flood	Tornado
1998	Bonnie, Charley, Earl, Frances, Georges, Hermine, Mitch	x	x	x	x	x
1999	Bret, Dennis, AL07, Floyd, Harvey, Irene	x	x	x	x	x
2000	AL04, Beryl, AL09, Gordon, Helene, Leslie	x	x	x	x	x
2001	Allison, Barry, Gabrielle, Karen, Michelle	x	x	x	x	x
2002	Arthur, Bertha, Cristobal, Edouard, Fay, Gustav, Hanna, Isidore, Kyle, Lili	x	x	x	x	x
2003	Bill, Claudette, AL07, Erika, Grace, Henri, Isabel	x	x	x	x	x
2004	Alex, Bonnie, Charley, Frances, Gaston, Hermine, Ivan, Jeanne, Matthew	x	x	x	x	x
2005	Arlene, Cindy, Dennis, Emily, Katrina, Ophelia, Rita, Tammy, Twenty-Two, Wilma	x	x	x	x	x
2006	Alberto, Beryl, Chris, Ernesto	x	x	x	x	x
2007	Andrea, Barry, Erin, Gabrielle, Humberto, Ten, Noel	x	x	x	x	x
2008	Cristobal, Dolly, Edouard, Fay, Gustav, Hanna, Ike, Kyle, Paloma	x	x	x	x	x
2009	One, Claudette, Ida	x	x	x	x	x
2010	Alex, Two, Bonnie, Five, Earl, Hermine, Nicole, Paula	x	x	x	x	x
2011	Bret, Don, Emily, Irene, Lee	x	x	x	x	x
2012	Alberto, Beryl, Debby, Isaac, Sandy	x	x		x	x
2013	Andrea, Dorian, Karen	x	x		x	x
2014	Arthur	x	x		x	x
2015	Ana, Bill, Claudette	x	x		x	x
2016	Bonnie, Colin, Eight, Hermine, Julia, Matthew	x	x		x	x
2017	Cindy, Emily, Harvey, Irma, Jose, Nate, Philippe	x	x		x	x
2018	Alberto, Chris, Florence, Gordon, Michael	x	x		x	x

In functions throughout the `hurricaneexposure` package, storms are identified based on their name and year (e.g., Hurricane Floyd in 1999 is identified as “Floyd-1999”). Note that a few storms satisfied the criteria to be included in the data but were unnamed (e.g., “AL13” in 1988). These storms are identified in functions in the `hurricaneexposure` package based on the identifying name listed in the above table, which is based on the first four characters of the storm’s US ATCF code for the storm. For some research projects, however, you may want to consider excluding these unnamed storms from the analysis.

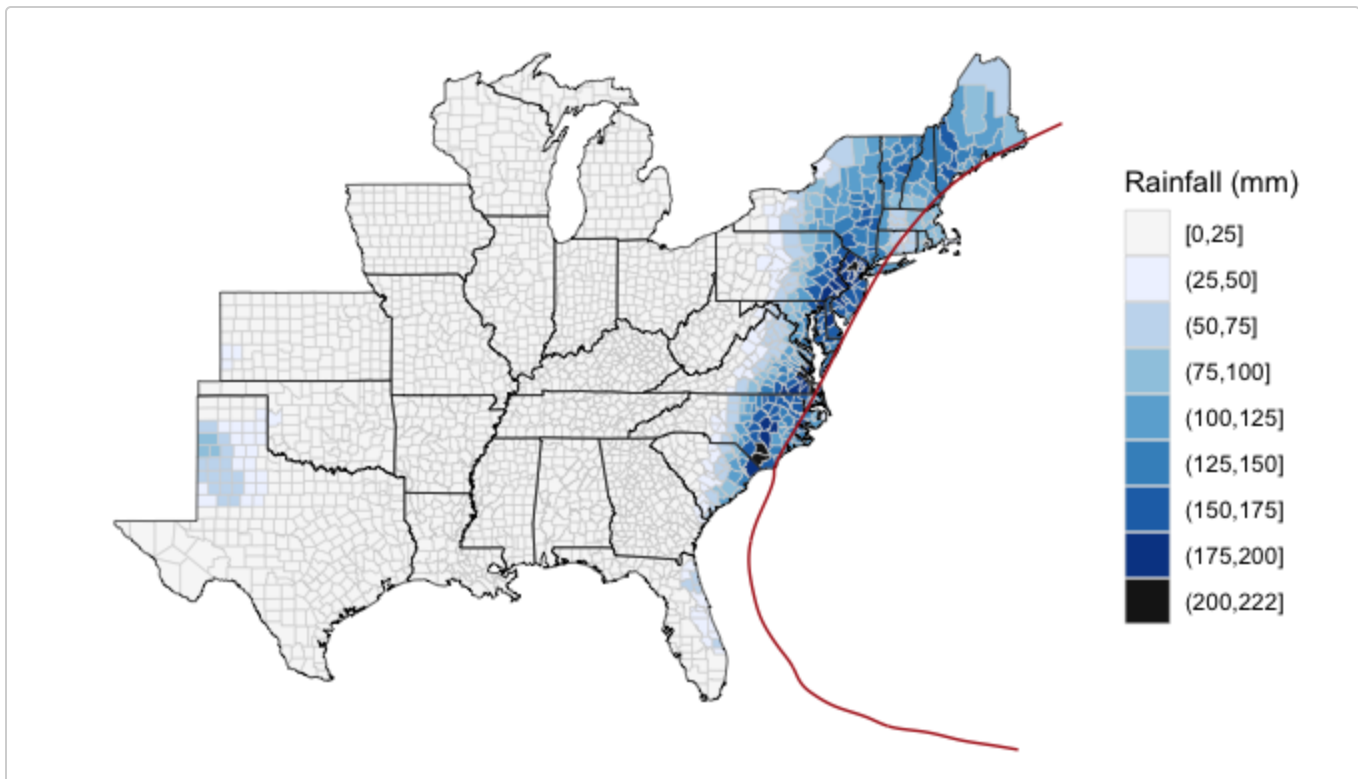
Mapping hurricane exposure

The `hurricaneexposure` package has functions to interact with the data in the `hurricaneexposuredata` package. First, there are functions that can be used to create some different maps of hurricane exposures based on distance to the storm track and rainfall.

Plotting county-level continuous exposure

For several of the storm hazard measurements (currently rain, wind, and distance), you can use the `map_counties` function in `hurricaneexposure` to plot a map for a specific storm showing the continuous value of that metric for all eastern U.S. counties. For example, you can map county-level rain during Hurricane Floyd (1999) using the following call:

```
map_counties(storm = "Floyd-1999", metric = "rainfall")
```



By default, this map shows the cumulative rain in each county for two days before to one day after the date that the storm passed closest to the county (see the help file for the `closest_dist` dataset in `hurricaneexposuredata` to find out more about how this date of closest approach was determined for each county-storm combination). The period used to calculate cumulative rainfall for the map can be customized using the `days_included` option in the `map_counties` function. The two following examples show the difference in rain maps for Tropical Storm Allison (2001) when using rain only from the day before and the day the storm was closest to each county (`days_included = -1:0`) versus from five days before to three days after the storm's closest approach (`days_included = -5:3`):

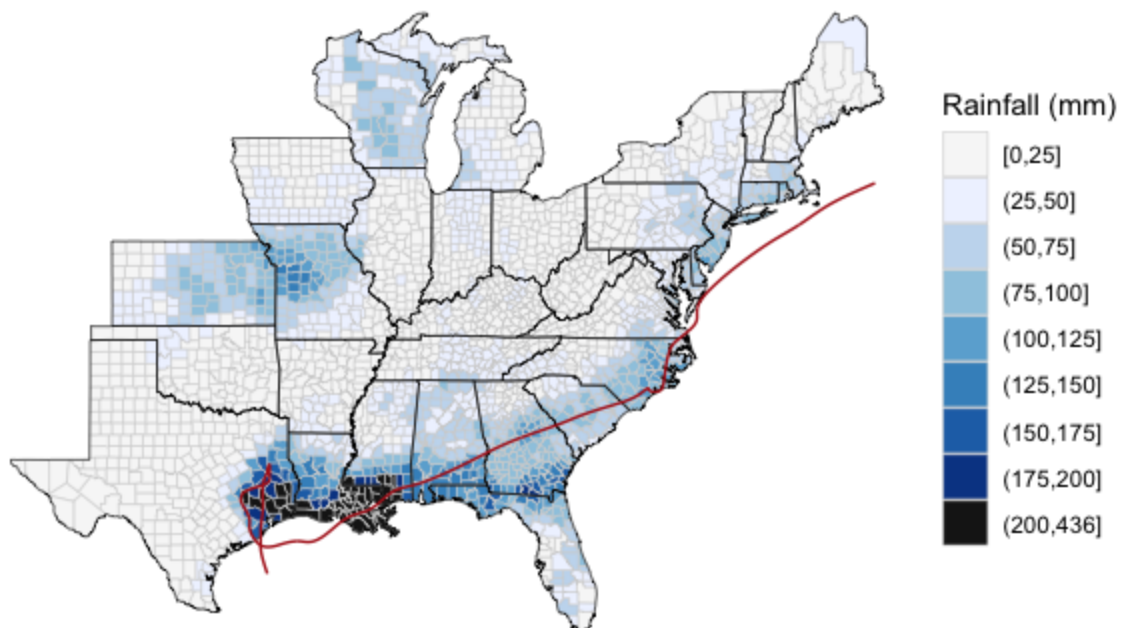
```
map_counties(storm = "Allison-2001", metric = "rainfall", days_included = -1:0) +  
  ggplot2::ggtitle("Rain during Allison (2001) for day before and day of closest approach")
```

Rain during Allison (2001) for day before and day of closest approach



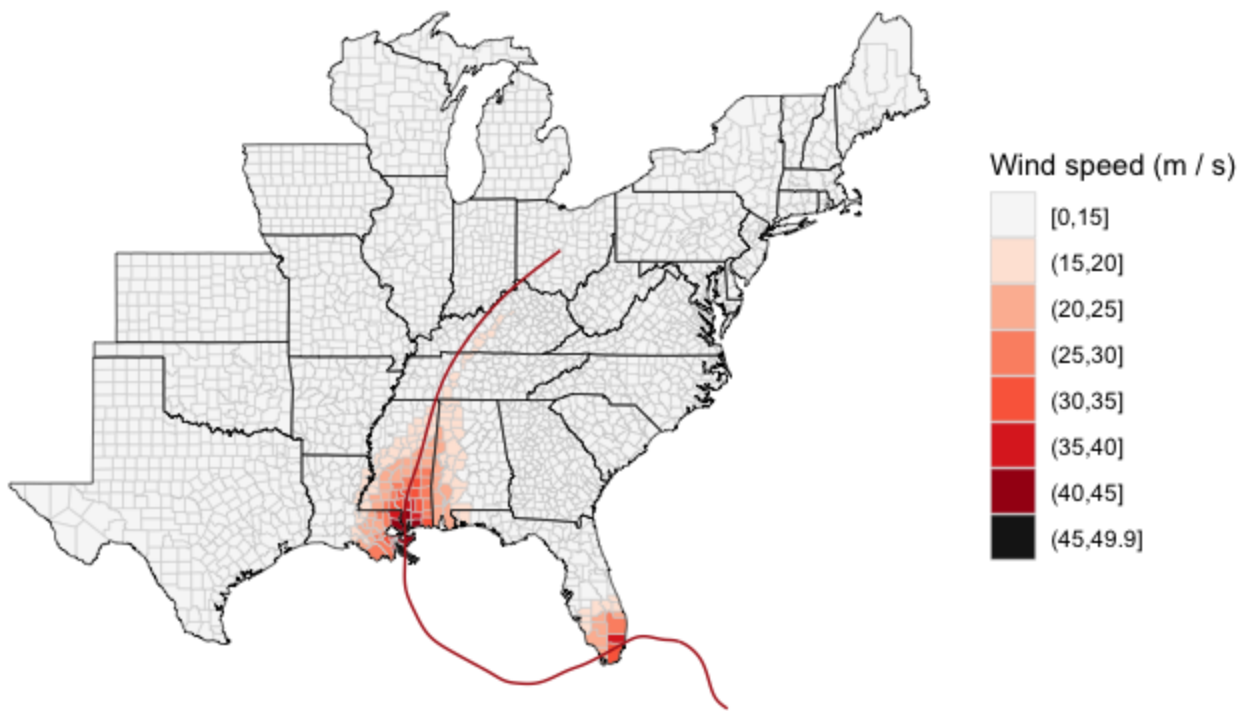
```
map_counties(storm = "Allison-2001", metric = "rainfall", days_included = -5:3) +  
  ggplot2::ggtitle("Rain during Allison (2001) for five days before to three days\nafter the day  
of closest approach")
```

Rain during Allison (2001) for five days before to three days after the day of closest approach



You can also use the `map_counties` function to plot the maximum wind during the storm for each county. For this, you use the argument `metric = "wind"`. For example, you can plot county-specific storm winds during Hurricane Katrina (2005) with the call:

```
map_counties(storm = "Katrina-2005", metric = "wind")
```



These county winds are determined based on a model of wind speeds, using as input data from the hurricane tracks data. See the documentation for the `storm_winds` dataset (`?storm_winds`) to find out more about this data; further details on the wind model are available in the vignettes for the [stormwindmodel package](#).

You can map a few other variables of wind with this function. For example, you can map the duration of winds of 20 m / s or more by using the argument `wind_var = "sust_dur"`:

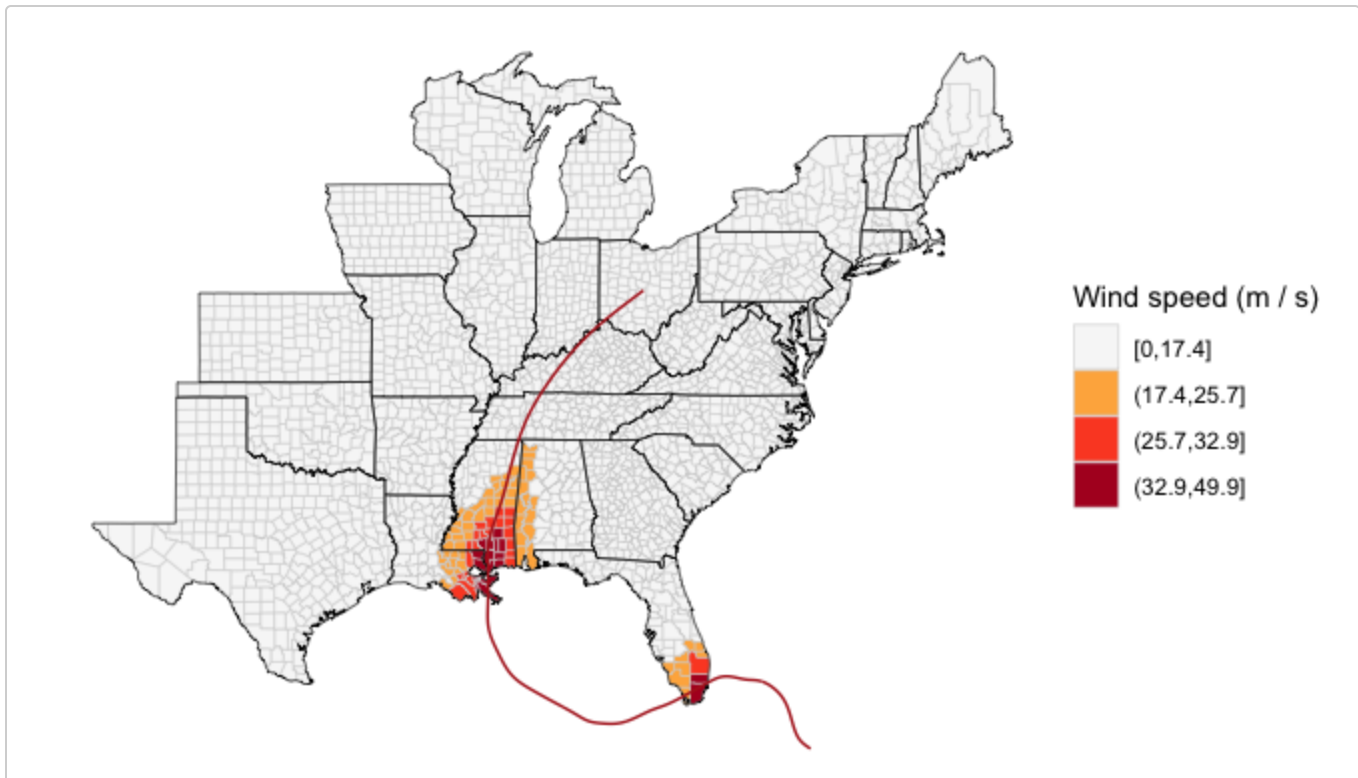
```
map_counties("Katrina-2005", metric = "wind", wind_var = "sust_dur")
```



You can map estimated gust winds, rather than sustained winds (the default), using the argument `wind_var = "vmax_gust"`. Further, we have included a second source of estimated winds in the data available in `hurricaneexposedata`. These wind estimates are based on the wind radii in the HURDAT2 dataset (you can find out more about this data in the helpfile for the `ext_tracks_wind` dataset). These data provide estimates

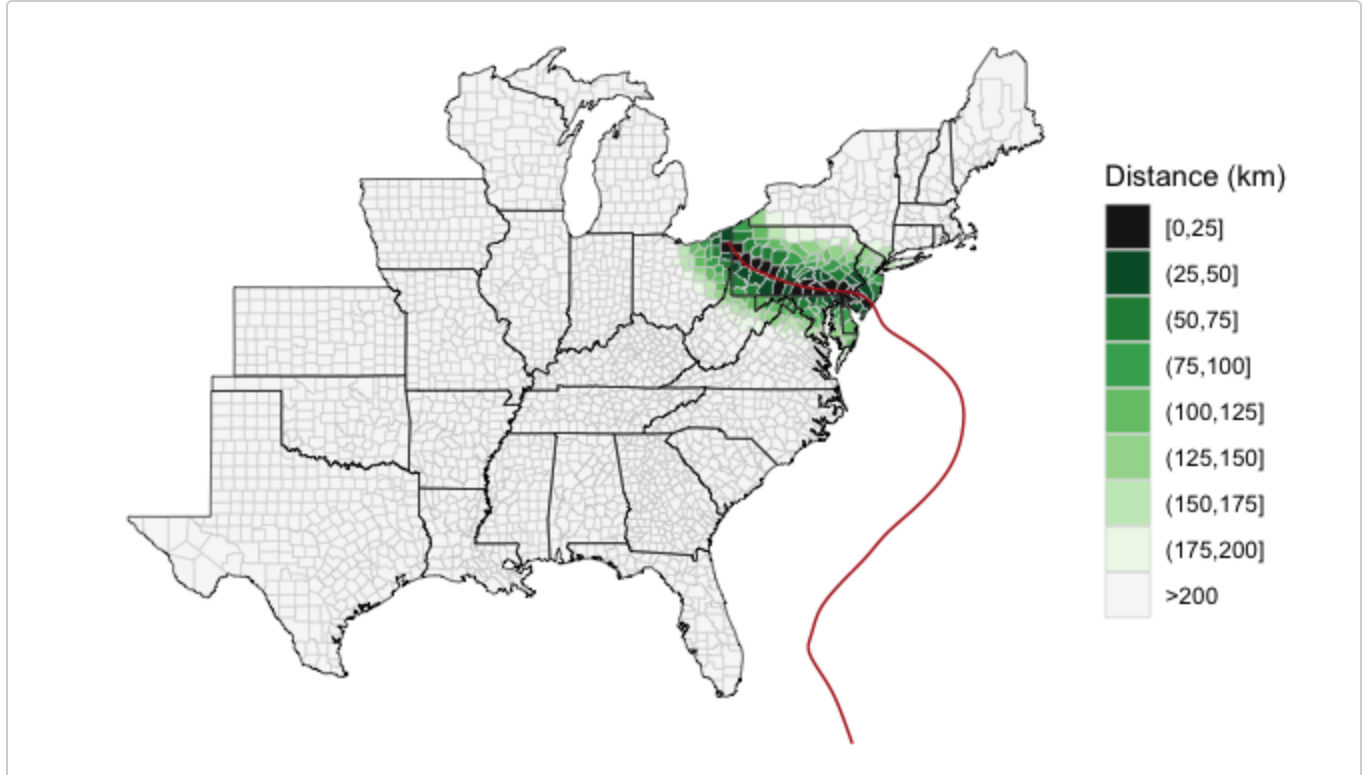
of which counties were exposed during a storm to sustained winds in four categories: 0–34 knots; 34–50 knots; 50–64 knots; and 64 knots or higher. Therefore, these data provide a categorical rather than continuous estimate of county wind speeds. However, they may be preferable to the model wind speeds for some storms, especially storms in extra-tropical transition. Further, you may find it interesting to use these data in a sensitivity analysis, to compare if impact study results are sensitive to whether these wind data or the modeled wind data are used for exposure classification. You can create a wind map based on this data using the `wind_source = "ext_tracks"` argument in the `map_counties` function. For example, to map estimated winds during Hurricane Katrina based on the HURDAT2 wind radii, you can run:

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



Finally, you can also use the `map_counties` function to plot the closest distance between the storm and each county. For this, you use the argument `metric = "distance"`. For example, the following call plots county-level distances from the track of Hurricane Sandy (2012):

```
map_counties(storm = "Sandy-2012", metric = "distance")
```



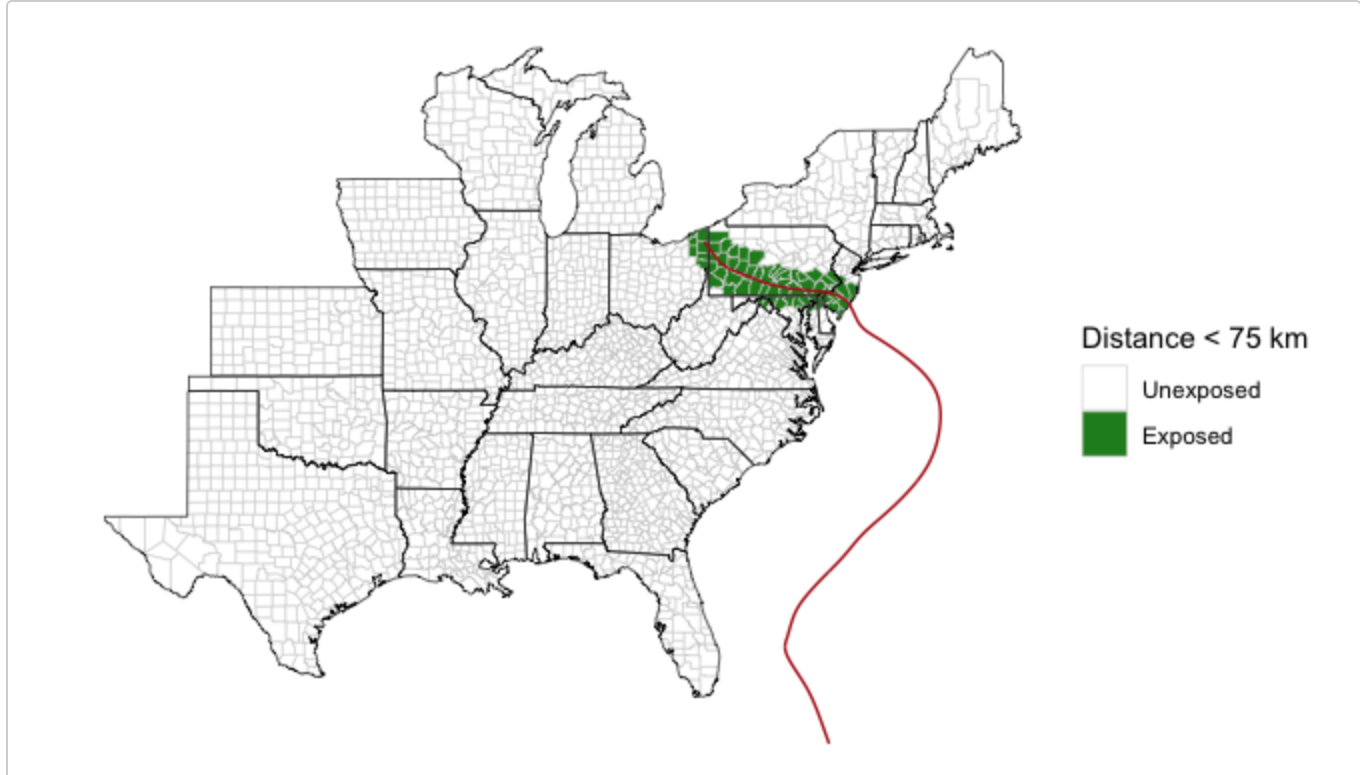
See the documentation for the `closest_dist` dataset (`?closest_dist`) for more details on how distances from the storm track were calculated for each county.

Mapping county-level binary exposure

You can also map binary county-level exposures, where each county is classified as “exposed” based on whether it meets a threshold for one of the hazard-based metrics. These maps can be created for the metrics shown in the continuous exposure maps above (wind, rain, and distance from the storm track). In addition, binary classification maps can also be created based on storm event listings for flood events and tornado events from the NOAA Storm Events database.

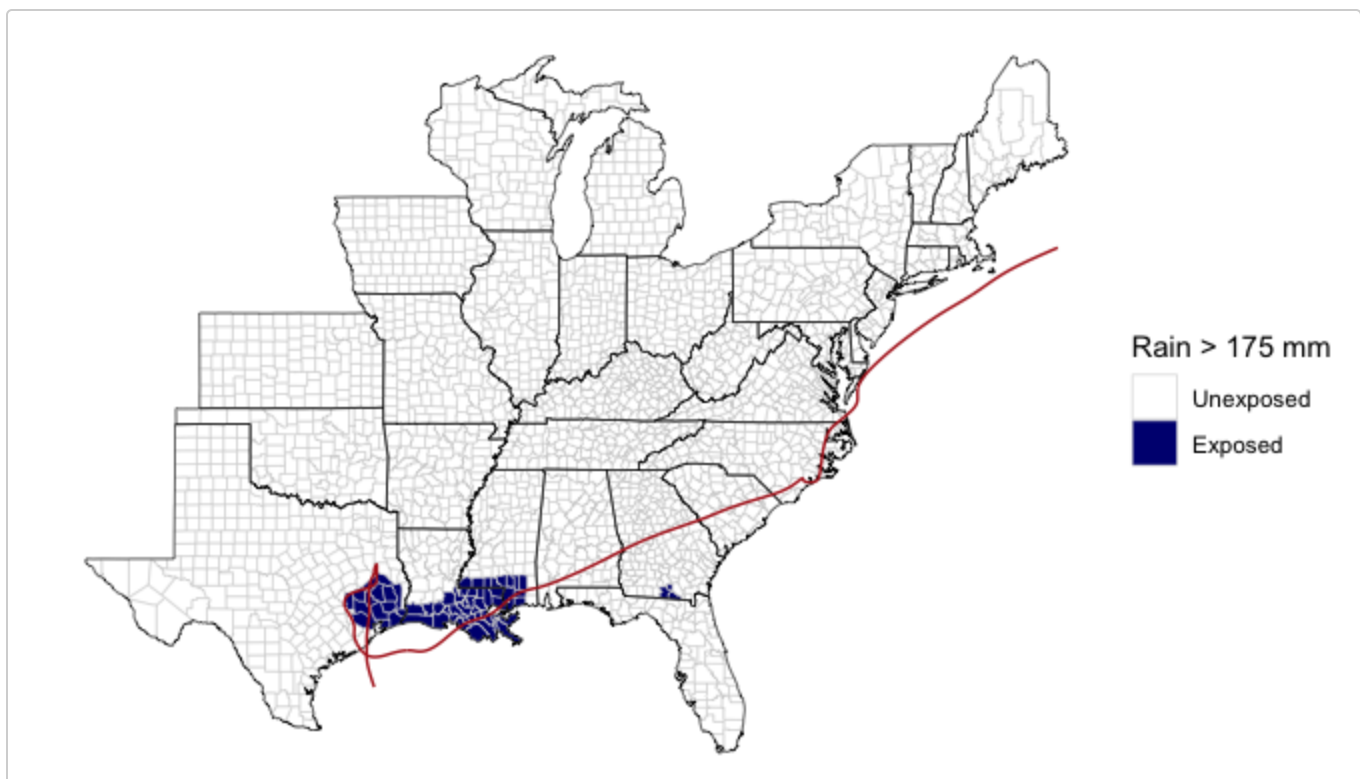
For example, you can map a binary variable of distance-based exposure using `map_distance_exposure`, with “exposure” defined as that the county was within the number of kilometers given in `dist_limit` of the storm track. For example, you can map the counties that were within 75 km of Hurricane Sandy’s storm track with the call:

```
map_distance_exposure(storm = "Sandy-2012", dist_limit = 75)
```

Similarly, you can map binary county exposure based on rain using `map_rain_exposure`. In the case of rain, you must define exposure thresholds for both rainfall and distance. This is because the rain data is observed data, so neglecting to limit “exposed” counties to those close to the storm track may pick up counties far from the storm that experienced a lot of rain over the same period. For example, you can use the following call to map rain exposure to Tropical Storm Allison (2001), where a county is defined as exposed to the storm if cumulative rainfall from five days before to three days after the storm’s closest approach was 175 mm or more and the county was within 500 km of the storm’s track:

```
map_rain_exposure(storm = "Allison-2001", rain_limit = 175, dist_limit = 500,  
                  days_included = -5:3)
```

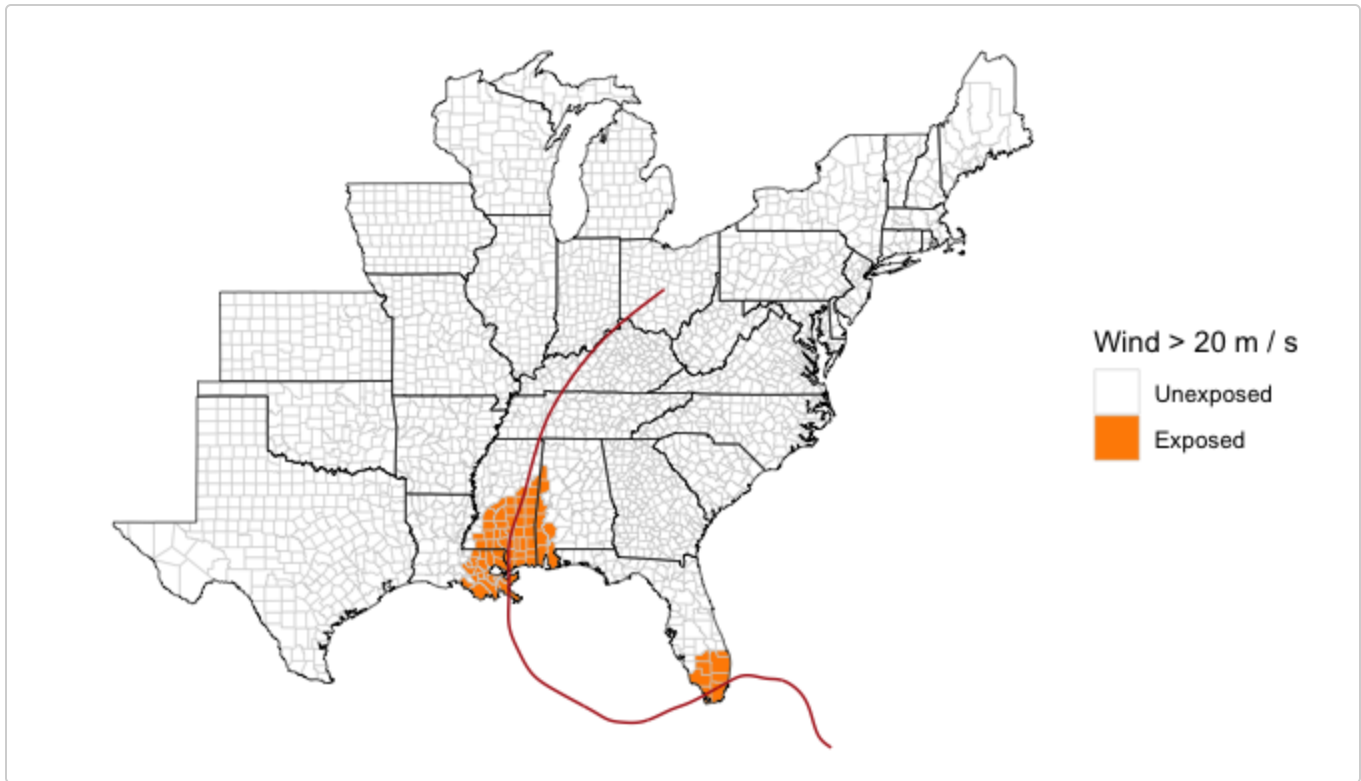


As with the continuous maps of rain exposure, the number of days included to calculate cumulative rain can be adjusted with the `days_included` option in `map_rain_exposure` (see the help file for `map_rain_exposure` for

more details).

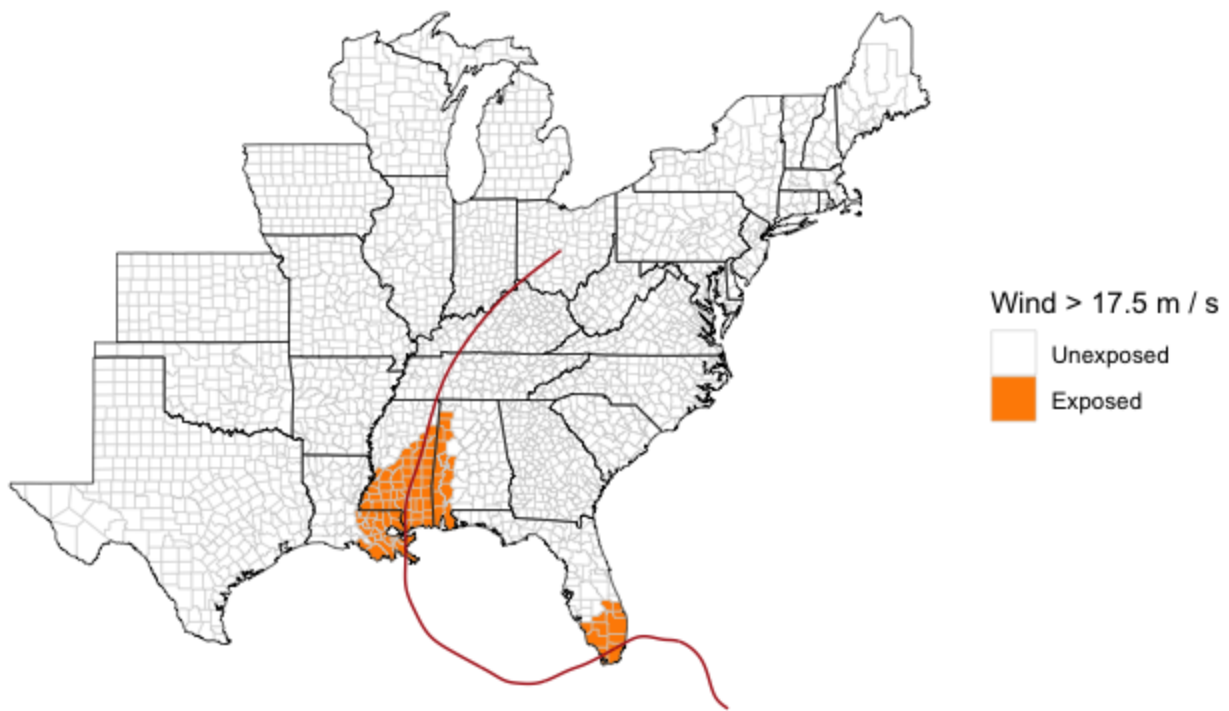
You can use the `map_wind_exposure` function to map counties exposed to a certain winds during a storm. For example, to identify all counties exposed to sustained winds of 20 m / s or higher during Hurricane Katrina, you can run:

```
map_wind_exposure(storm = "Katrina-2005", wind_limit = 20)
```



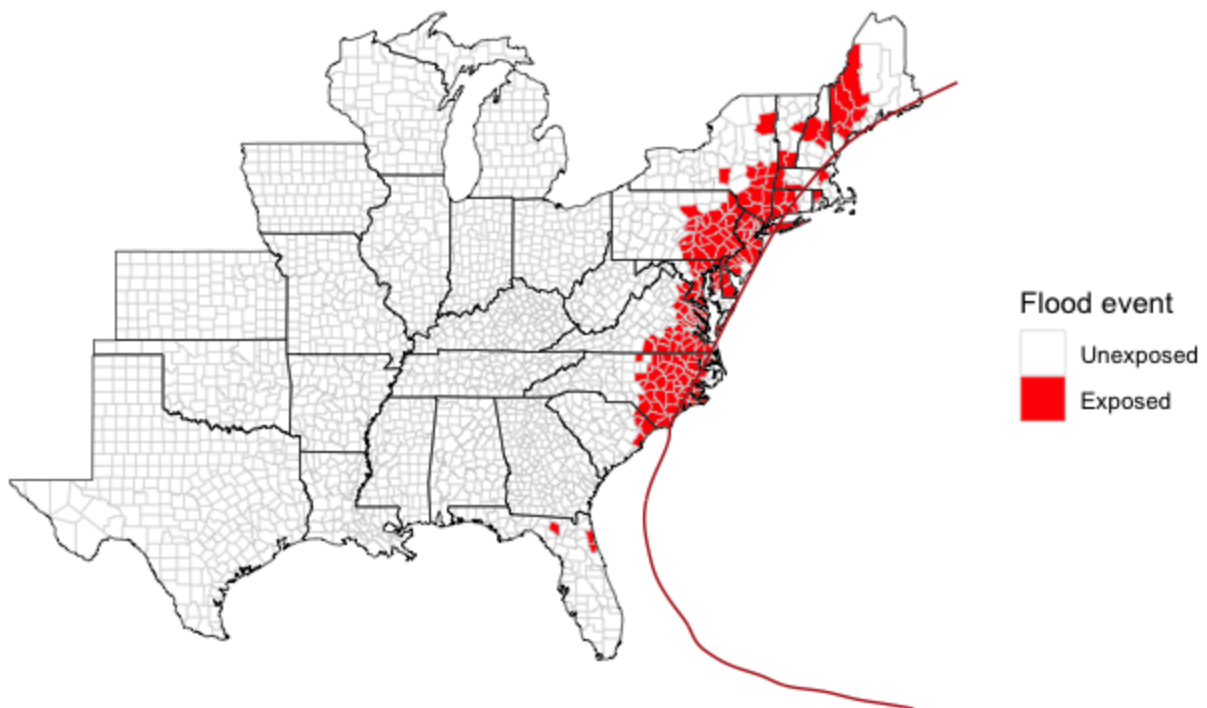
If you would like to base a wind threshold on knots rather than m / s, you can use the `convert_wind_speed` function from the `weathermetrics` package to do that. For example, to map counties in which winds were 34 knots or higher during the storm, you can run:

```
library(weathermetrics)
map_wind_exposure(storm = "Katrina-2005",
  wind_limit = convert_wind_speed(34, "knots", "mps"))
```



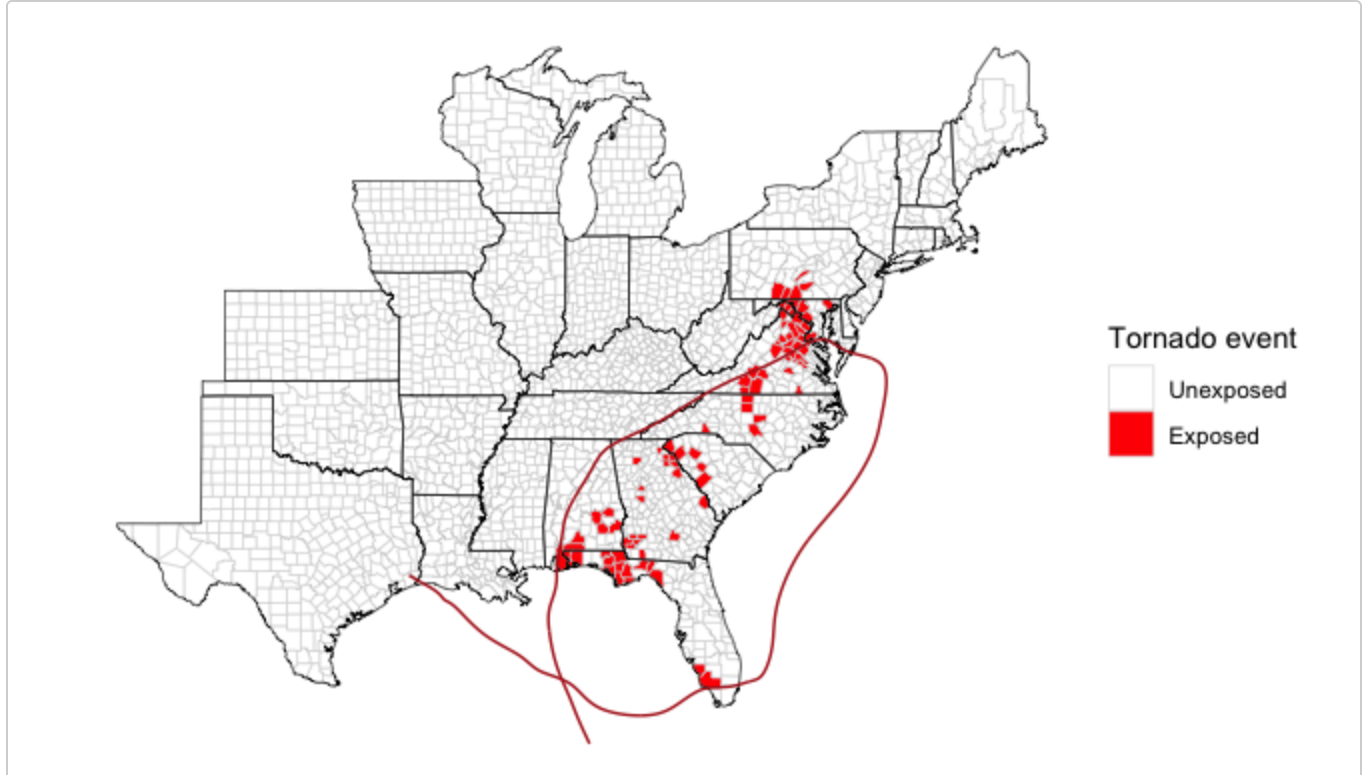
Finally, you can map which counties were exposed to specific types of events, as listed in the NOAA Storm Events Database, using the `map_event_exposure` function. For example, you can map counties for which a flood event was listed for Hurricane Floyd (1999) with the call:

```
map_event_exposure(storm = "Floyd-1999", event_type = "flood")
```



Similarly, you can map which counties were exposed to tornado events during Hurricane Ivan (2004) with:

```
map_event_exposure(storm = "Ivan-2004", event_type = "tornado")
```



When using this function, note that the types of events reported in the NOAA Storm Events Database changed in 1996, so for many event types there will be no listings before 1996. Tornado listings do extend earlier, but flood listings do not. Further, when using this data, the user should be aware of and take into account the limitations of the database, especially for comparing storms across years. Reporting standards likely have changed over time; for example, it is possible that earlier years have more false negatives than more recent years.

Event listings from the NOAA Storm Events database were linked to storms based on the start date of the event being within a five-day window of the date the storm was closest to the county and the storm coming within 500 km of the county. The [noaastormevents package](#) was used to create this dataset. For more information on the data, see the help file for the `storm_events` dataset (`?storm_events`) or the vignette for the `noaastormevents` package.

Plotting storm tracks

If you would like to add a storm's track to one of the maps of county-level exposure, you can do so using the `map_tracks` function. This function can be used either to map a storm track on a blank U.S. map or to add a track to an existing map, if the map is a `ggplot` object (as are all maps generated in the previous section).

First, the `map_tracks` function can be used to create a map of the hurricane tracks for one or more storms. For example, to plot the tracks of Hurricane Floyd in 1999, you can run:

```
map_tracks(storms = "Floyd-1999")
```



There are some different options you can use for the tracks' appearance. For example, if you wanted to plot the tracks of several storms, and plot each point when the track locations were measured (typically every six hours), use some transparency so you can see all the tracks, and show the tracks in blue, you can run:

```
map_tracks(storms = c("Andrew-1992", "Katrina-2005", "Rita-2005"),  
           alpha = 0.5, plot_points = TRUE, color = "blue")
```



As another example, to map all tracks for storms in 2018, you can run:

```
library(dplyr)  
library(tidyr)  
storms_2018 <- hurr_tracks %>%
```

```

select(storm_id) %>%
distinct() %>%
mutate(year = stringr::str_extract(storm_id, "-[0-9].+")) %>%
filter(year == "-2018")
map_tracks(storms = storms_2018$storm_id)

```

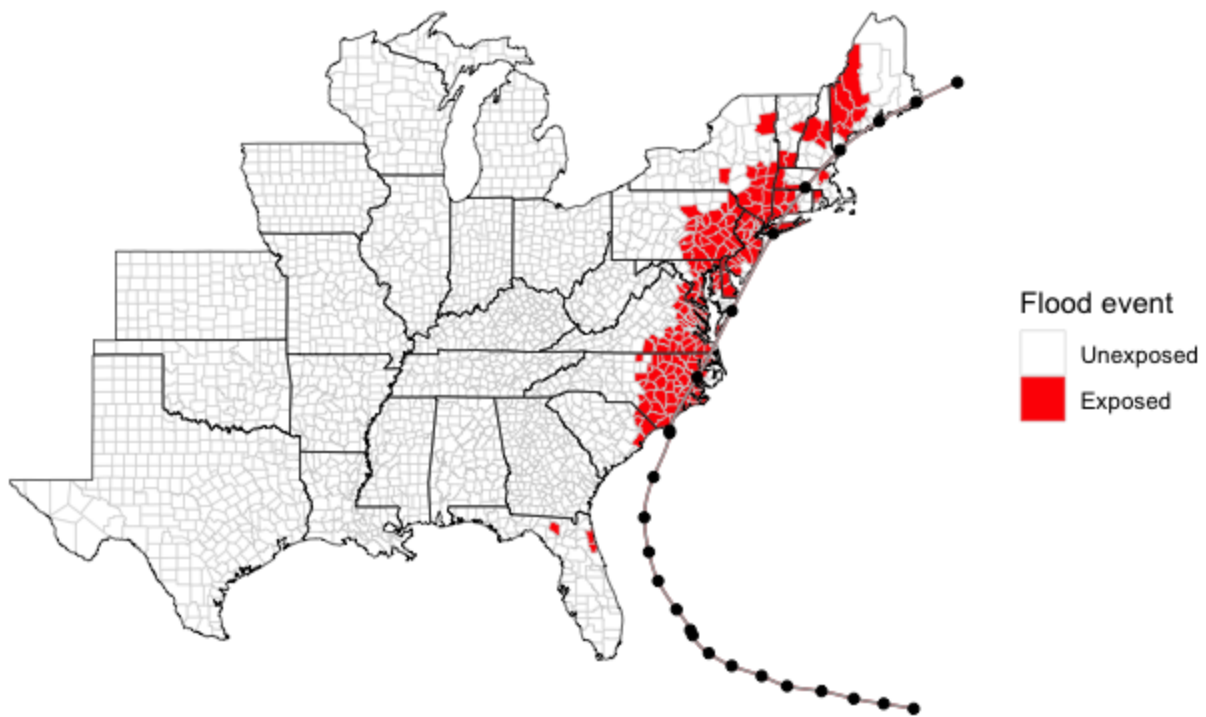


The `map_tracks` function is used in all the mapping functions described in the previous section, to add the storm's track to the exposure maps. However, if you would like to customize the appearance of the storm's track on the map, you can do so by plotting the exposure map without the storm track (`add_track = FALSE`) and then adding the track yourself with the `map_tracks` function. For example, to change the color of the storm track in a map of flood event exposure during Hurricane Floyd (1999) and to add points showing the available observations for the storm in the Best Tracks hurricane tracking data, you can run:

```

floyd_map <- map_event_exposure(storm = "Floyd-1999", event_type = "flood")
map_tracks(storms = "Floyd-1999", plot_object = floyd_map, plot_points = TRUE,
           color = "darkgray")

```

Creating time series datasets of exposure

The `hurricaneexposure` package also has several functions that can input a list of counties and output a list of all of the storms to which each county was exposed over a certain period, based on one of the hazard metrics. These exposure listings include the storm dates, making them easy to integrate into time series datasets of health or other impact data.

The `county_rain` function takes a list of county FIPS codes, bounds on the starting and ending years of the analysis, and thresholds to define rain-based exposure (cumulative rainfall and distance from the storm's track, as explained for the binary rain maps shown in an earlier section) and creates a list of all storms that met these thresholds for the counties. For example, to get a dataset of all the storms to which Orleans Parish (FIPS 22071), and Newport News, Virginia (FIPS 51700), were exposed between 1995 and 2005, where "exposed" means that the storm passed within 100 kilometers of the county center and the rainfall over a three-day window of the date of closest approach was 100 millimeters or more, you can run:

```
county_rain(counties = c("22071", "51700"), start_year = 1995, end_year = 2005,
            rain_limit = 100, dist_limit = 100, days_included = c(-1, 0, 1))
```

```
##      storm_id  fips closest_date storm_dist tot_precip      local_time
## 1:   Bill-2003 22071   2003-06-30   47.74045    141.1 2003-06-30 18:15
## 2: Charley-2004 51700   2004-08-14   55.58752    136.2 2004-08-14 19:45
## 3:   Cindy-2005 22071   2005-07-06   26.62602    113.2 2005-07-06 02:15
## 4:   Floyd-1999 51700   1999-09-16   49.43163    207.5 1999-09-16 11:15
## 5: Isidore-2002 22071   2002-09-26   14.27431    249.0 2002-09-26 06:45
## 6: Katrina-2005 22071   2005-08-29   44.07046    196.2 2005-08-29 09:00
## 7: Matthew-2004 22071   2004-10-10   74.88275    123.2 2004-10-10 10:00
##      closest_time_utc
## 1: 2003-06-30 23:15
## 2: 2004-08-14 23:45
## 3: 2005-07-06 07:15
## 4: 1999-09-16 15:15
## 5: 2002-09-26 11:45
```

```
## 6: 2005-08-29 14:00
## 7: 2004-10-10 15:00
```

This function draws on the same dataset (`rain`) from the `hurricaneexposedata` package as the functions for mapping rain exposure shown above. For this and other exposures, see the table earlier in this vignette to see which years of data are available (for rain, data is currently available for storms through 2011).

In addition to giving the names and closest dates of each storm for each county (`closest_date`—note, this is given based on local time for the county; see documentation for the [countytimezones](#) package to find out more about how local time was calculated from UTC for each county), this function also gives you the distance between the county and the storm's track at the time when the storm was closest to the county's population weighted center (`storm_dist`, in kilometers) and the total precipitation over the included days (`tot_precip`). The returned dataframe also gives the time of closest approach in both local time (`local_time`) and UTC (`closest_time_utc`), based on 15-minute intervals along the storm's track.

Similar functions are available in the package to create listings of county exposures to storms based on wind, distance to the storm track, and events listings from the NOAA Storm Events Database (flood and tornado events). For example, to get a listing of all storms between 1988 and 2015 for which Miami-Dade county (FIPS:) has experienced sustained winds of 34 knots or more (17.5 m / s), you can run:

```
county_wind(counties = "12086", start_year = 1988, end_year = 2015, wind_limit = 17.5)
```

```
##      storm_id  fips vmax_sust vmax_gust sust_dur gust_dur usa_atcf_id
## 1  Andrew-1992 12086  56.60869  84.34695     615     960    AL041992
## 2   Mitch-1998 12086  18.79781  28.00873       0     285    AL131998
## 3   Harvey-1999 12086  23.01816  34.29706      75     285    AL101999
## 4    Irene-1999 12086  31.88419  47.50744     615    1290    AL131999
## 5  Frances-2004 12086  20.77793  30.95911     285    1830    AL062004
## 6   Jeanne-2004 12086  20.53852  30.60239      90    1065    AL112004
## 7  Katrina-2005 12086  33.14848  49.39124     525    1740    AL122005
## 8    Wilma-2005 12086  33.32450  49.65351     315     600    AL252005
## 9  Ernesto-2006 12086  18.02811  26.86188       0     765    AL062006
## 10   Fay-2008 12086  18.85232  28.08996       0    1305    AL062008
## closest_time_utc storm_dist      local_time closest_date
## 1 1992-08-24 09:00   30.48032 1992-08-24 05:00 1992-08-24
## 2 1998-11-05 12:45  130.19217 1998-11-05 07:45 1998-11-05
## 3 1999-09-21 20:45   48.90460 1999-09-21 16:45 1999-09-21
## 4 1999-10-15 23:15   43.07201 1999-10-15 19:15 1999-10-15
## 5 2004-09-05 02:30  156.03624 2004-09-04 22:30 2004-09-04
## 6 2004-09-26 03:15  156.99903 2004-09-25 23:15 2004-09-25
## 7 2005-08-26 00:30   12.33865 2005-08-25 20:30 2005-08-25
## 8 2005-10-24 12:45   76.55817 2005-10-24 08:45 2005-10-24
## 9 2006-08-30 09:15   69.47981 2006-08-30 05:15 2006-08-30
## 10 2008-08-19 10:15  127.08899 2008-08-19 06:15 2008-08-19
```

The returned dataframe includes the estimated maximum values during the storm for sustained winds (`vmax_sust`) and gust winds (`vmax_gust`), both in m / s, as well as the duration for which sustained and gust winds were over 20 m / s in the county (`sust_dur` and `gust_dur`, respectively). It also includes some of the variables on the date and time of closest approach and the distance at the storm's closest approach.

The `county_wind` function allows some further specifications when identifying the storms to which a county was exposed based on wind estimates. First, it allows the use of alternative wind variables. While the default is to use the maximum sustained wind speed experienced in the county during the storm, it is also possible to base the exposure metric on the maximum gust winds (`wind_var = "vmax_gust"`) or on the duration of either sustained or gust winds at or above 20 m / s (`wind_var = "sust_dur"` and `wind_var = "gust_dur"`, respectively). For example, to determine which storms brought Orleans Parish sustained winds of 34 knots or higher (17.5 m / s or higher) for an hour or more (`wind_limit = 60`; for wind durations, this wind limit should be specified in minutes), you can run:

```
county_wind(counties = "12086", start_year = 1988, end_year = 2015,
            wind_var = "sust_dur", wind_limit = 60)
```

```
##      storm_id  fips vmax_sust vmax_gust sust_dur gust_dur usa_atcf_id
## 1 Andrew-1992 12086  56.60869  84.34695     615     960   AL041992
## 2 Harvey-1999 12086  23.01816  34.29706      75     285   AL101999
## 3 Irene-1999 12086  31.88419  47.50744     615    1290   AL131999
## 4 Frances-2004 12086  20.77793  30.95911     285    1830   AL062004
## 5 Jeanne-2004 12086  20.53852  30.60239      90    1065   AL112004
## 6 Katrina-2005 12086  33.14848  49.39124     525    1740   AL122005
## 7 Wilma-2005 12086  33.32450  49.65351     315     600   AL252005
## closest_time_utc storm_dist      local_time closest_date
## 1 1992-08-24 09:00   30.48032 1992-08-24 05:00   1992-08-24
## 2 1999-09-21 20:45   48.90460 1999-09-21 16:45   1999-09-21
## 3 1999-10-15 23:15   43.07201 1999-10-15 19:15   1999-10-15
## 4 2004-09-05 02:30  156.03624 2004-09-04 22:30   2004-09-04
## 5 2004-09-26 03:15  156.99903 2004-09-25 23:15   2004-09-25
## 6 2005-08-26 00:30   12.33865 2005-08-25 20:30   2005-08-25
## 7 2005-10-24 12:45   76.55817 2005-10-24 08:45   2005-10-24
```

Further, the `county_wind` function allows you to pull wind estimates from an alternate source. By default, this function uses wind estimates for each county from a wind model (see the vignettes for the `stormwindmodel` package for much more detail on this modeling process). However, you can also pull estimates based on the wind radii from HURDAT2 by using the option `wind_source = "ext_tracks"`. See the help file for the `ext_tracks_wind` dataset that comes with `hurricaneexposedata` for more details on this data source and how the county-specific exposures were determined from the data source for use in the `county_wind` function. It is important to note that this data results in wind estimates with breaks at 34 knots, 50 knots, and 64 knots, rather than continuous estimates. This means that a sustained wind estimate from this data source of 17.4896 m / s (34 knots) is estimating that the county had maximum wind speeds of 34 knots or higher during the storm, but not as high as 50 knots. Further, the `dur_sust` value when using this data source is based on number of minutes with winds at or above 34 knots (rather than the 20 m / s value used for durations for the modeled wind data). Gust durations can not be determined using the HURDAT2 wind radii.

The HURDAT2 wind radii are more tied to observations during a specific storm than the modeled data. In certain cases, especially storms in extratropical transition, this dataset might be preferable to the modeled wind speeds, even though it provides less continuous estimates. This source of wind data can be specified using the `wind_source` option in the `county_wind` function. For example, to generate a list of all storms in Orleans Parish with maximum sustained winds of 34 knots or more based on the HURDAT2 wind radii, you can run:

```
county_wind(counties = "12086", start_year = 1988, end_year = 2015,
            wind_var = "vmax_sust", wind_limit = 17.4, wind_source = "ext_tracks")
```

```
##      storm_id  fips vmax_sust vmax_gust sust_dur usa_atcf_id closest_time_utc
## 1 Bonnie-2010 12086  17.4896  26.05950     180   AL032010 2010-07-23 15:30
## 2 Ernesto-2006 12086  17.4896  26.05950     855   AL062006 2006-08-30 09:15
## 3 Fay-2008 12086  17.4896  26.05950    1230   AL062008 2008-08-19 10:15
## 4 Frances-2004 12086  17.4896  26.05950    1455   AL062004 2004-09-05 02:30
## 5 Isaac-2012 12086  17.4896  26.05950     720   AL092012 2012-08-26 15:15
## 6 Jeanne-2004 12086  17.4896  26.05950     570   AL112004 2004-09-26 03:15
## 7 Katrina-2005 12086  32.9216  49.05318     720   AL122005 2005-08-26 00:30
## 8 Wilma-2005 12086  32.9216  49.05318     795   AL252005 2005-10-24 12:45
## storm_dist      local_time closest_date
## 1  27.25331 2010-07-23 11:30   2010-07-23
## 2  69.47981 2006-08-30 05:15   2006-08-30
```

```
## 3 127.08899 2008-08-19 06:15 2008-08-19
## 4 156.03624 2004-09-04 22:30 2004-09-04
## 5 248.34174 2012-08-26 11:15 2012-08-26
## 6 156.99903 2004-09-25 23:15 2004-09-25
## 7 12.33865 2005-08-25 20:30 2005-08-25
## 8 76.55817 2005-10-24 08:45 2005-10-24
```

The `county_distance` function can similarly be used to generate a listing of all storms that came within a certain distance of specified counties. For example, to get a list of all storms that came within 50 km of Orleans Parish (FIPS 22071) between 1988 and 2015, you can run:

```
county_distance(counties = "22071", start_year = 1988, end_year = 2015,
               dist_limit = 50)
```

```
##      storm_id fips closest_date storm_dist      local_time
## 1:  Beryl-1988 22071 1988-08-09 3.379783 1988-08-09 04:45
## 2: Florence-1988 22071 1988-09-10 4.217016 1988-09-10 03:00
## 3: Allison-2001 22071 2001-06-11 23.289781 2001-06-11 02:30
## 4:  Bertha-2002 22071 2002-08-05 16.190337 2002-08-05 03:45
## 5:  Isidore-2002 22071 2002-09-26 14.274314 2002-09-26 06:45
## 6:    Bill-2003 22071 2003-06-30 47.740448 2003-06-30 18:15
## 7:   Cindy-2005 22071 2005-07-06 26.626022 2005-07-06 02:15
## 8: Katrina-2005 22071 2005-08-29 44.070459 2005-08-29 09:00
## 9:  Bonnie-2010 22071 2010-07-25 40.631881 2010-07-25 05:30
## closest_time_utc
## 1: 1988-08-09 09:45
## 2: 1988-09-10 08:00
## 3: 2001-06-11 07:30
## 4: 2002-08-05 08:45
## 5: 2002-09-26 11:45
## 6: 2003-06-30 23:15
## 7: 2005-07-06 07:15
## 8: 2005-08-29 14:00
## 9: 2010-07-25 10:30
```

Finally, you can use the `county_events` function to get a listing of all storms for which a county had a certain type of NOAA Storm Events listing. For example, to get a list of all storms for which Norfolk, Virginia, (FIPS: 51710) had a flood event listing, you can run:

```
county_events(counties = "51710", start_year = 1996, end_year = 2015,
              event_type = "flood")
```

```
##      fips      storm_id usa_atcf_id closest_time_utc storm_dist
## 1 51710  Arthur-1996    AL011996 1996-06-20 08:00 164.49133
## 2 51710    AL01-1997    AL011997 1997-06-01 18:30 246.23488
## 3 51710  Dennis-1999    AL051999 1999-09-05 05:00 200.44453
## 4 51710  Floyd-1999    AL081999 1999-09-16 15:00 20.91885
## 5 51710  Irene-1999    AL131999 1999-10-18 07:30 234.49601
## 6 51710    Kyle-2002    AL122002 2002-10-12 08:15 149.04476
## 7 51710    Alex-2004    AL012004 2004-08-03 18:45 199.38999
## 8 51710  Alberto-2006   AL012006 2006-06-14 20:15 105.73478
## 9 51710  Ernesto-2006   AL062006 2006-09-01 23:45 69.60457
## 10 51710  Irene-2011    AL092011 2011-08-28 00:00 54.49080
## 11 51710  Sandy-2012    AL182012 2012-10-29 20:30 289.06525
## 12 51710 Claudette-2015 AL032015 2015-07-12 00:00 182.77412
##      local_time closest_date
```

```
## 1 1996-06-20 04:00 1996-06-20
## 2 1997-06-01 14:30 1997-06-01
## 3 1999-09-05 01:00 1999-09-05
## 4 1999-09-16 11:00 1999-09-16
## 5 1999-10-18 03:30 1999-10-18
## 6 2002-10-12 04:15 2002-10-12
## 7 2004-08-03 14:45 2004-08-03
## 8 2006-06-14 16:15 2006-06-14
## 9 2006-09-01 19:45 2006-09-01
## 10 2011-08-27 20:00 2011-08-27
## 11 2012-10-29 16:30 2012-10-29
## 12 2015-07-11 20:00 2015-07-11
```

Handling multi-county communities

Some datasets covering storm impacts might have communities composed of multiple counties rather than county-specific listings. To get a dataframe listing the relevant storms for multi-county communities, you can use the `multi_county_*` family of functions in a similar way to the `county_*` family of functions. For example, to get rain exposure listings in the case where one of the communities (New York, NY) is comprised of multiple counties, you can run:

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

```
## # A tibble: 10 x 9
## # Groups:   community_name [3]
##   community_name storm_id closest_date local_time closest_time_utc mean_dist
##   <fct>          <chr>    <chr>      <chr>      <chr>                <dbl>
## 1 new           Charley... 2004-08-14 2004-08-1... 2004-08-14 23:45      55.6
## 2 new           Danny-1... 1997-07-24 1997-07-2... 1997-07-24 19:00      61.9
## 3 new           Floyd-1... 1999-09-16 1999-09-1... 1999-09-16 15:15      49.4
## 4 no            Allison... 2001-06-11 2001-06-1... 2001-06-11 07:30      23.3
## 5 no            Bill-20... 2003-06-30 2003-06-3... 2003-06-30 23:15      47.7
## 6 no            Cindy-2... 2005-07-06 2005-07-0... 2005-07-06 07:15      26.6
## 7 no            Isidore... 2002-09-26 2002-09-2... 2002-09-26 11:45      14.3
## 8 no            Katrina... 2005-08-29 2005-08-2... 2005-08-29 14:00      44.1
## 9 no            Matthew... 2004-10-10 2004-10-1... 2004-10-10 15:00      74.9
## 10 ny           Floyd-1... 1999-09-16 1999-09-1... 1999-09-17 00:15      41.2
## # ... with 3 more variables: mean_rain <dbl>, max_rain <dbl>, min_dist <dbl>
```

The output from this function includes columns for the average closest distance for any of the counties in the community (`mean_dist`), the average precipitation for all the counties (`mean_precip`), the highest precipitation for any of the counties (`max_rain`), and the smallest distance between the storm track and any of the county population-weighted centers (`min_dist`).

Similar functions exist for wind exposure (`multi_county_wind`) and distance exposure (`multi_county_distance`). In some cases, these multi-county functions may have less functionality than their `county_*` counterparts. For example, the `multi_county_wind` function currently only allows use of the sustained wind speeds for the exposure metric, while the `county_wind` function allows the use of sustained winds, gust winds, or durations of either.