Output Data

Dawn Nekorchuk, Michael Wimberly, and EPIDEMIA Team Members
Department of Geography and Environmental Sustainability, University of Oklahoma
dawn.nekorchuk@ou.edu; mcwimberly@ou.edu

Updated May 09, 2019

Contents

1 Epidemiar Output Dataset - Full Run

The results of `run_epidemia()` is a named list of the following datasets or objects:

1. `summary_data`
2. `epi_summary`
3. `modeling_results_data`
4. `environ_timeseries`
5. `environ_anomalies`
6. `params_meta`
7. `regression_object`

1.1 `summary_data`

Early detection and early warning alerts levels for each geographic group. Early detection alerts (`ed_alert_count`) are alerts that are triggered during the early detection period, which was defined by the argument `ed_summary_period`, the n most recent weeks of known epidemiology data. Similarly, early warning alerts (`ew_alert_count`) are alerts in the future forecast estimates. "High" level indicates two or more weeks in this period had incidences greater than the alert threshold, "Medium" means that one week was in alert status, and "Low" means no weeks had alerts (`ed_sum_level` and `ew_level`, respectively).

- `{groupfield}`: The user-given geographic grouping field
- `ed_alert_count`: Number of alerts triggered in the early detection period
- `ed_sum_level`: High/Medium/Low depending on the number of alerts, 2+/1/0 respectively
- `ew_alert_count`: Number of alerts triggered in the early warning period (future forecast period)
- `ew_level`: High/Medium/Low depending on the number of alerts, 2+/1/0 respectively

1.2 `epi_summary`

Mean disease incidence per geographic group during the early detection period.

- `{groupfield}`: The user-given geographic grouping field
- `mean_inc`: The mean disease incidence per geographic group summarized over the early detection period

1.3 `modeling_results_data`

These are multiple timeseries values for observed, forecast, and alert thresholds of disease incidence, over the report period, for each geographic unit. These data can be used in creating the individual geographic unit control charts.

- `{groupfield}`: The user-given geographic grouping field
- `obs_date`: The last day of the epidemiological week (ISO/CDC, by `week_type`), Date object
- `series`: "obs" = observed disease incidence, "fc" = modeled/forecast incidence values, "thresh" = event detection threshold values, "ed" = early detection alert (binary), "ew" = early warning alert (binary)
- `value`: Value of the `series` for that geographic group for that week
- `lab`: Labels for the series ("Observed", "Forecast Trend", "Alert Threshold", "Early Detection Alert", "Early Warning Alert")
- `upper`: Unused
- `lower`: Unused
- `week_epidemiar`: ISO/CDC week number, based on user given `week_type` argument
- `year_epidemiar`: ISO/CDC year, based on user given `week_type` argument

1.4 `environ_timeseries`

These are multiple timeseries for the used environmental variables during the report period for each geographic unit.

- `{groupfield}`: The user-given geographic grouping field
- `{obsfield}`: The user-given field for the environmental variable name/ID
- `year_epidemiar`: ISO/CDC year, based on user given `week_type` argument
- `week_epidemiar`: ISO/CDC week number, based on user given `week_type` argument
- `obs_date`: The last day of the epidemiological week (ISO/CDC, by `week_type`), Date object
- `val_epidemiar`: Value of the environmental variable for that geographic group for that week. Values are a combination of observed, or interpolated (for missing) or extended (future estimated) values.

- `reference_method`: Method for creating a weekly summary from daily data (e.g. "sum" for rainfall, or "mean" for NDWI)
- `data_source`: "Observed", "Interpolated", or "Extended". Missing environmental data is handled in three different ways, depending on time period. For missing values in the middle of series, the value is a linear approximation of surrounding values ("Interpolated"). For missing values at the end of the series, up to the future forecast portion, values are carried forward in a persistence approach (also marked "Interpolated" at the moment). For the forecast future portion, values are a blending of the last known values and the climatic historical mean, with a gradual weighting scheme shifting from more weight from last known to historical mean ("Extended").

- `ref_value`: From `env_ref_data`.

- `ref_*`: Fields from `env_ref_data` that begin with `ref_` have been propogating through to here. (Potentially useful for plotting, for example.)

1.5 `environ_anomalies`

These data are the recent (during the early detection period) differences (anomalies) of the environmental variable values from the climatology/reference mean.

- `{groupfield}`: The user-given geographic grouping field

- `{obsfield}`: The user-given field for the environmental variable name/ID
- `anom_ed_mean`: The mean of the anomalies per environmental variable per geographic group summarized during the early detection period. The anomalies are calculated as the difference from the observed value to the historical mean for that week of the year.

1.6 `params_meta`

This lists dates, settings, and parameters that `run_epidemiar()` was called with.

1.7 `regression_object`

This is the regression object from the general additive model (bam()) regression. This is generally only for additional statistical investigation of the model, and is usually not saved (large object).

2 Epidemiar Output Dataset - Model Only Run

The results of `run_epidemiar(..., model_run = TRUE)` is only the regression object (model) plus some metadata information about what was used to generate the model. Once a model has been generated, it can be fed back into `run_epidemiar(..., model_obj = {regression object})` for faster predictions rather than regenerating the model on each run. Determining the balance on how old of a model is still useful is heavily dependent on the specific dataset.

1. `model_obj`
2. `model_info`

2.1 `model_obj`

The output regression object from the `bam()` general additive model regression call.

2.2 `model_info`

A list of dates, settings, and relevant parameters that `run_epidemiar()` was called with. Very similar to `params_meta` of a full run.