How to verify submission files

Nutcha Wattanachit, Nicholas Reich

19 March, 2020

This vignette explains how to verify submission files for the CDC COVID-19 ILI challenge. Submission files must follow specific formatting guidelines.

There are 2 ways to verify submission files using the `cdcForecastUtils` package. First, if you have not installed the `cdcForecastUtils` package, then please do so using `devtools`:

```r
devtools::install_github("reichlab/cdcForecastUtils")
```

When this package is installed, you can load it in to your current session:

```r
library(cdcForecastUtils)
```

Then, choose how to verify using the methods in the examples below.

Verify with `verify_entry_file()`

This method verifies using a file path without saving the forecasts in the R environment.

```r
# For national/Regional-level forecasts
weekly_forecast_filepath1 <- "/directory/EW10-2020-regional_forecast_file.csv"
verify_entry_file(weekly_forecast_filepath1)

# For state-level forecasts
weekly_forecast_filepath2 <- "/directory/EW10-2020-state_forecast_file.csv"
verify_entry_file(weekly_forecast_filepath2, challenge = "state_ili")
```

Verify with `verify_entry()`

This method is almost the same as the previous method, except the forecasts are read into R environment with `read_entry()`, which the user can inspect if the verification were to fail.

```r
# For national/Regional-level forecasts
weekly_forecast_file1 <- read_entry("/directory/EW10-2020-regional_forecast_file.csv")
verify_entry(weekly_forecast_file1)

# For state-level forecasts
```

```
weekly_forecast_file2 <- read_entry("/directory/EW10-2020-state_forecast_file.csv")
verify_entry(weekly_forecast_file2)
```

For both methods, there will be no errors if the forecast file is formatted correctly. Please refer to the template for more details regarding formatting. Note that there could be warnings or/and messages for missing locations or/and missing targets, which can be ignored if the forecasts for those targets or/and locations are intentionally not included.