Package 'synopR'

May 9, 2026

Title Fast Decoding of SYNOP (Surface Synoptic Observations) Meteorological Messages

Version 1.0.0

Description Decode raw SYNOP (surface synoptic observations) messages into data frames, extracting data from Sections 0, 1, and 3, including temperature, dew point, pressure, wind, clouds, and precipitation. Available functions to download SYNOP messages from Ogimet <https://www.ogimet.com/> if needed. The decoding logic follows the specifications defined in the World Meteorological Organization (2019) ``Manual on Codes, Volume I.1 (WMO-No. 306)".

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Suggests tibble, knitr, rmarkdown, testthat (>= 3.0.0)

URL https://ezequiel1593.github.io/synopR/

BugReports https://github.com/ezequiel1593/synopR/issues

VignetteBuilder knitr

NeedsCompilation no

Author Ezequiel Elias [aut, cre]

Maintainer Ezequiel Elias <elezeru15@gmail.com>

Repository CRAN

Date/Publication 2026-04-03 21:50:02 UTC

Contents

check_synop *Check SYNOP messages for structural integrity*

Description

Validates if SYNOP strings meet basic structural requirements, considering section indicators and 5-digit data groups.

Usage

```
check_synop(data)
```

Arguments

data	A character vector of SYNOP strings or the exact data frame returned by `parse_ogimet()`.

Value

A data frame with validation results for each message.

Examples

```
msg <- paste0("AAXX 01123 87736 32965 13205 10214 20143 ",
              "30022 40113 5//// 80005 333 10236 20128=")
checked_synops <- check_synop(msg)
```

direct_download_from_ogimet *Direct download of meteorological data from Ogimet*

Description

Direct download of meteorological data from Ogimet

Usage

```
direct_download_from_ogimet(wmo_identifier, initial_date, final_date)
```

Arguments

wmo_identifier	A 5-digit character string or integer representing the station WMO ID.
initial_date	Initial date, format "YYYY-MM-DD".
final_date	Final date, format "YYYY-MM-DD".

Details

The requested period cannot exceed 370 days. All queries assume UTC time zone. The returned data frame covers from 00:00 UTC of the `initial_date` to 23:00 UTC of the `final_date`, inclusive. Too many requests may trigger temporal blocks.

If the station identifier starts with 0 (zero), then `wmo_identifier` must be a string (e.g., "06447").

Value

A data frame, as returned by `show_synop_data()`

Examples

```
## Not run:
direct_download_from_ogimet(wmo_identifier = '87585',
                            initial_date = "2024-01-10",
                            final_date = "2024-01-11")

## End(Not run)
```

`download_from_ogimet` *Download SYNOP messages from Ogimet*

Description

Download SYNOP messages from Ogimet

Usage

```
download_from_ogimet(wmo_identifier, initial_date, final_date)
```

Arguments

`wmo_identifier`	A 5-digit character string or integer representing the station WMO ID.
`initial_date`	Initial date, format "YYYY-MM-DD".
`final_date`	Final date, format "YYYY-MM-DD".

Details

The requested period cannot exceed 370 days. All queries assume UTC time zone. The returned dataset covers from 00:00 UTC of the `initial_date` to 23:00 UTC of the `final_date`, inclusive. Too many requests may trigger temporal blocks.

If the station identifier starts with 0 (zero), then `wmo_identifier` must be a string (e.g., "06447").

Value

A character vector with SYNOP strings.

Examples

```
## Not run:
download_from_ogimet(wmo_identifier = '87585',
                     initial_date = "2024-01-10",
                     final_date = "2024-01-11")

## End(Not run)
```

`parse_ogimet` *Parse SYNOP strings downloaded from Ogimet into a data frame*

Description

Parse SYNOP strings downloaded from Ogimet into a data frame

Usage

```
parse_ogimet(ogimet_data)
```

Arguments

`ogimet_data`	A character vector of Ogimet-format SYNOP strings.

Value

A data frame with Year, Month, Day, Hour, and Raw_synop.

Examples

```
msg <- paste0("87736,2026,01,01,12,00,AAXX 01123 87736 32965 13205 10214 20143 ",
              "30022 40113 5//// 80005 333 10236 20128=")
parsed_data <- parse_ogimet(msg)
```

`show_synop_data` *Decode multiple SYNOP messages*

Description

This function decodes a vector or data frame column of SYNOP strings belonging to the same or different meteorological surface station.

Usage

```
show_synop_data(data, wmo_identifier = NULL, remove_empty_cols = TRUE)
```

Arguments

`data`	A character vector, a data frame column containing raw SYNOP strings, or the exact data frame returned by `parse_ogimet()`.
`wmo_identifier`	A 5-digit character string or integer representing the station WMO ID. If NULL (default), all messages are decoded.
`remove_empty_cols`	Logical. Should columns containing only `NA` values be removed? Default is TRUE.

Value

A data frame where each row represents one observation time and each column a decoded meteorological variable.

1. wmo_id - WMO station identifier
2. Year - (from parse_ogimet())
3. Day - As informed by Section 0
4. Hour - As informed by Section 0
5. Cloud_base_height - Lowest cloud base height, in intervals
6. Visibility - In meters
7. Total_cloud_cover - In oktas, 9 means 'invisible' sky by fog or other phenomenon
8. Wind_direction - In tens of degree, 99 means 'variable wind direction'
9. Wind_speed
10. Wind_speed_unit - Either 'm/s' or 'knots'
11. Air_temperature - In degrees Celsius
12. Dew_point - In degrees Celsius
13. Relative_humidity - As a percentage
14. Station_pressure - In hPa
15. MSLP_GH - Mean sea level pressure (in hPa) or geopotential height (in gpm)
16. Pressure_tendency - In hPa
17. Charac_pressure_tend - String, simplified decoding
18. Precipitation_S1 - In mm
19. Precip_period_S1 - In hours ('Precipitation_S1' fell in the last 'Precip_period_S1' hours)
20. Present_weather - String, simplified decoding
21. Past_weather1 - String, simplified decoding
22. Past_weather2 - String, simplified decoding
23. Cloud_amount_Nh - Cloud coverage from low or medium cloud, same as 'Total_cloud_cover'
24. Low_clouds_CL - String, simplified decoding
25. Medium_clouds_CM - String, simplified decoding
26. High_clouds_CH - String, simplified decoding

27. Max_temperature - In degrees Celsius
28. Min_temperature - In degrees Celsius
29. Ground_state - String, simplified decoding
30. Ground_temperature - Integer, in degrees Celsius
31. Snow_ground_state - String, simplified decoding
32. Snow_depth - In cm
33. Ev_Evt - Evaporation (ev) or evapotranspiration (evt), in mm
34. Sunshine_daily - In hours (generally from the previous civil day)
35. Positive_Net_Rad_last_24h - In J/cm^2
36. Negative_Net_Rad_last_24h - In J/cm^2
37. Global_Solar_Rad_last_24h - In J/cm^2
38. Diffused_Solar_Rad_last_24h - In J/cm^2
39. Downward_LongWave_Rad_last_24h - In J/cm^2
40. Upward_LongWave_Rad_last_24h - In J/cm^2
41. ShortWave_Rad_last_24h - In J/cm^2
42. Net_ShortWave_Rad_last_24h - In J/cm^2
43. Direct_Solar_Rad_last_24h - In J/cm^2
44. Sunshine_last_hour - In hours
45. Positive_Net_Rad_last_hour - In kJ/m^2
46. Negative_Net_Rad_last_hour - In kJ/m^2
47. Global_Solar_Rad_last_hour - In kJ/m^2
48. Diffused_Solar_Rad_last_hour - In kJ/m^2
49. Downward_LongWave_Rad_last_hour - In kJ/m^2
50. Upward_LongWave_Rad_last_hour - In kJ/m^2
51. ShortWave_Rad_last_hour - In kJ/m^2
52. Net_ShortWave_Rad_last_hour - In kJ/m^2
53. Direct_Solar_Rad_last_hour - In kJ/m^2
54. Cloud_drift_direction - In cardinal and intercardinal directions for "low - medium - high" clouds
55. Cloud_elevation_direction - String indicating genera, direction and elevation angle
56. Pressure_change_last_24h - In hPa
57. Precipitation_S3 - In mm
58. Precip_period_S3 - In hours ('Precipitation_S3' fell in the last 'Precip_period_S3' hours)
59. Precipitation_last_24h - In mm
60. Cloud_layer_1 - String indicating cover, genera and height
61. Cloud_layer_2 - String indicating cover, genera and height
62. Cloud_layer_3 - String indicating cover, genera and height
63. Cloud_layer_4 - String indicating cover, genera and height

Examples

```
msg <- paste0("AAXX 01123 87736 32965 13205 10214 20143 ",
              "30022 40113 5//// 80005 333 10236 20128 56000 81270=")
synop_df <- data.frame(messages = msg)
decoded_data <- show_synop_data(synop_df)
```

Index