

# Package ‘tpm’

June 1, 2024

**Title** FHWA TPM Score Calculation Functions

**Version** 2.0.2

**Description** Contains functions for calculating the Federal Highway Administration (FHWA) Transportation Performance Management (TPM) performance measures. Currently, the package provides methods for the System Reliability and Freight (PM3) performance measures calculated from travel time data provided by The National Performance Management Research Data Set (NPMRDS), including Level of Travel Time Reliability (LOTTR), Truck Travel Time Reliability (TTTR), and Peak Hour Excessive Delay (PHED) metric scores for calculating statewide reliability performance measures. Implements <[https://www.fhwa.dot.gov/tpm/guidance/pm3\\_hpms.pdf](https://www.fhwa.dot.gov/tpm/guidance/pm3_hpms.pdf)>.

**License** Mozilla Public License Version 2.0

**Encoding** UTF-8

**URL** [https://github.com/markegge/fhwa\\_pm3/](https://github.com/markegge/fhwa_pm3/)

**BugReports** [https://github.com/markegge/fhwa\\_pm3/issues](https://github.com/markegge/fhwa_pm3/issues)

**LazyData** true

**Depends** R (>= 3.5.0)

**Imports** data.table (>= 1.13), fasttime

**RoxygenNote** 7.3.1

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Mark Egge [aut, cre] (<<https://orcid.org/0009-0007-9128-2099>>)

**Maintainer** Mark Egge <mark@eateggs.com>

**Repository** CRAN

**Date/Publication** 2024-06-01 08:20:02 UTC

## R topics documented:

fips_lookup . . . . .	2
hpms . . . . .	2
lottr . . . . .	3
phed . . . . .	4
score . . . . .	6
tpm . . . . .	7
ttr . . . . .	7

<b>Index</b>	<b>9</b>
--------------	----------

---

fips_lookup	<i>FIPS Codes</i>
-------------	-------------------

---

### Description

This dataset provides a crosswalk between state names, postal codes, and FIPS codes. Used by the `hpms()` function. The variables are as follows:

### Usage

```
fips_lookup
```

### Format

A data frame with 56 rows and 4 variables:

**State\_Name** state name (e.g. 'Alabama')

**Postal\_Code** two character state postal code (e.g. 'AL')

**FIPS\_Code** Census Bureau FIPS Code (e.g. 1)

**STATE\_NAME** capitalized state name (e.g. 'ALAMABA')

---

hpms	<i>Generate an HPMS Submission File</i>
------	---

---

### Description

Generate an HPMS submission file in accordance with [HPMS Field Manual Supplemental Guidance](#) Requires the scores from `score()` to be run with `verbose = TRUE` The reporting year is based on the `TMC_Identification` year (e.g. use 2021 TMC network for 2021 reporting in 2022) Writes the resulting file to `hpms_year.txt`

**Usage**

```

hpms(
  file,
  tmc_identification,
  lottr_scores,
  tttr_scores,
  phed_scores = NULL,
  occ_fac = 1.7
)

```

**Arguments**

<code>file</code>	Output file name. This is the HPMS submittal file.
<code>tmc_identification</code>	Path to TMC_Identification.csv file provided by RITIS with travel time download
<code>lottr_scores</code>	A data.table of LOTTR scores produced using <code>score(..., metric == "LOTTR")</code>
<code>tttr_scores</code>	A data.table of TTTR scores produced using <code>score(..., metric == "TTTR")</code>
<code>phed_scores</code>	A data.table of of PHED scores produced using <code>phed()</code>
<code>occ_fac</code>	Occupancy factor. Default = 1.7

**Value**

No return value, writes file to disk

**Examples**

```

## Not run:
lottr_scores <- lottr("data/All_Vehicles/al_tt_seconds.csv", verbose = TRUE)
tttr_scores <- tttr("data/Trucks/aldot_2019_trucks.csv", verbose = TRUE)
phed_scores <- phed("Readings.csv", "TMC_Identification.csv",
  speed_limits = fread("speed_limits.csv"),
  urban_code = 56139, pm_peak = 3, population = 52898)
hpms("hpms_2020.txt", "TMC_Identification.csv", lottr_scores, tttr_scores, phed_scores)

## End(Not run)

```

---

lottr

*Calculate LOTTR Metric Score*


---

**Description**

Calculate LOTTR given a RITIS NPMRDS export of travel time data. Data is passed in as the path to the csv containing the travel time readings.\ Travel time units must be seconds and averaging should be 15 minutes.

**Usage**

```
lottr(travel_time_readings = NULL, monthly = FALSE, verbose = FALSE)
```

**Arguments**

travel_time_readings	path to RITIS export CSV with 15-minute average readings for all vehicles
monthly	TRUE or FALSE specifies if the results should be aggregated by month. If FALSE scores will be computed by TMC for all data in the input file.
verbose	Provide diagnostic output and return all calculated values (necessary for <a href="#">hpms</a> function)

**Value**

A data.table of LOTTR scores by TMC

**Examples**

```
## Not run:
lottr("data/All_Vehicles/Readings.csv")
lottr("data/All_Vehicles/Readings.csv", monthly = TRUE)

## End(Not run)
```

---

phed

*Calculate PHED Metric*

---

**Description**

Calculate the CMAQ Traffic Congestion Measure in accordance with [FHWA General Guidance and Step-by-Step Metric Calculation Procedures for National Performance Measures for Congestion, Reliability, and Freight, and CMAQ Traffic Congestion](#) Requires the speed limits for all TMC segments. Other parameters are optional and (with defaults supplied from FHWA's guidance). Uses the same travel time readings file as `score(..., metric = "LOTTR")`. Outputs annual hours of delay.

**Usage**

```
phed(
  travel_time_readings,
  tmc_identification,
  speed_limits,
  urban_code = NA,
  pm_peak = 3,
  avo_cars = 1.7,
  avo_trucks = 1,
```

```

    avo_buses = 10.7,
    moy_factor = moy_factor_default,
    dow_factor = dow_factor_default,
    hod_profile = hod_profile_default,
    population = NA
)

```

### Arguments

travel_time_readings	path to readings CSV with 15-minute travel time observations for all vehicles exported from RITIS.
tmc_identification	Path to TMC_Identification.csv file provided by RITIS with travel time download.
speed_limits	a data.frame-like object with speed limits for all TMCs with format tmc,speed_limit
urban_code	optional vector of one (or more) urban_code values. if specified, filters TMCs to the relevant urban_code
pm_peak	3 or 4. Peak Period is defined as weekdays from 6 am to 10 am and either 3 pm to 7 pm (3) or 4 pm to 8 pm (4)
avo_cars	Average vehicle occupancy for passenger vehicles
avo_trucks	Average vehicle occupancy for freight trucks
avo_buses	Average vehicle occupancy for buses
moy_factor	Month of year traffic adjustment factors for freeways and non-freeway facilities in format month, freeway, non_freeway 1, 0.99, 0.98 ... 12, 1.01, 1.00
dow_factor	Day of week adjustment factors. Monday (2) through Friday (6). Format: day, freeway, non_freeway 2, 1.05, 1.05 ... 6, 1.1, 1.1
hod_profile	Hourly traffic percentages for peak hours. Non-directional. Format: hour, freeway, non_freeway 6, 0.045, 0.050 ... 18, 0.052, 0.048
population	Optional population value. If provided, function will print PHED value

### Value

Annual hours of peak hour excessive delay per capita

## Examples

```
## Not run:
phed(travel_time_readings = "nprds/all_vehicles_2021/Readings.csv",
     tmc_identification = "nprds/all_vehicles_2021/TMC_Identification.csv",
     speed_limits = fread("birmingham_tmc_speed_limits.csv"),
     urban_code = 7786,
     pm_peak = 3,
     output_file_name = "phed_2021.csv",
     avo_cars = 1.62, avo_trucks = 1.0, avo_buses = 5.1,
     moy_factor = fread("birmingham_moy_factors.csv"),
     dow_factor = fread("birmingham_dow_factors.csv"),
     hod_profile = fread("birmingham_hod_profile.csv"),
     population = 752898)

## End(Not run)
```

---

score

*Internal function to Calculate LOTTR or TTTR Metric Score*

---

## Description

Calculate LOTTR / TTTR given a RITIS NPMRDS export of travel time data. Data is passed in as the path to the csv containing the travel time readings. input file must have header and format: tmc\_code,measurement\_tstamp,travel\_time\_seconds e.g. > tmc\_code,measurement\_tstamp,travel\_time\_seconds > 116-04379,2019-01-01 00:00:00,44.78 > 116-04379,2019-01-01 00:15:00,46.69

## Usage

```
score(input_file = NULL, metric, monthly = FALSE, verbose = FALSE)
```

## Arguments

input_file	Path to file containing travel time readings
metric	"LOTTR" or "TTTR"
monthly	TRUE or FALSE specifies if the results should be aggregated by month. If FALSE scores will be computed by TMC for all data in the input file.
verbose	Provide diagnostic output and return all calculated values

## Value

A data.table of LOTTR/TTTR scores by TMC

---

tpm	<i>TPM Tools: A package for calculating TPM PM3 Travel Time Reliability Scores from NPMRDS Data</i>
-----	---

---

### Description

This package will provides functions needed to calculate PM3 System Reliability and Freight and CMAQ Congestion Federal TPM Performance measures

### Details

Note: if your state has a large amount of data, you may encounter a "Error: vector memory exhausted (limit reached?)" error. See this StackOverflow post on resolving: <https://stackoverflow.com/questions/51295402/r-on-macos-error-vector-memory-exhausted-limit-reached>

### PM3 functions

The functions `lottr` calculated LOTTR metric scores for TMC segments `tttr` calculates TTTR metric scores for TMC segments `phed` calculates PHED metric scores for TMC segments `hpms` generates an HPMS submission file in pipe delimited format

---

tttr	<i>Calculate TTTR Metric Score</i>
------	------------------------------------

---

### Description

Calculate TTTR given a RITIS NPMRDS export of travel time data. Data is passed in as the path to the csv containing the travel time readings. Travel time units must be seconds and averaging should be 15 minutes.

### Usage

```
tttr(travel_time_readings = NULL, monthly = FALSE, verbose = FALSE)
```

### Arguments

travel_time_readings	path to RITIS export CSV with 15-minute average readings for trucks
monthly	TRUE or FALSE specifies if the results should be aggregated by month. If FALSE scores will be computed by TMC for all data in the input file.
verbose	Provide diagnostic output and return all calculated values (necessary for <code>hpms</code> function)

### Value

A data.table of TTTR scores by TMC

**Examples**

```
## Not run:  
ttr("data/Trucks/Readings.csv")  
ttr("data/Trucks/Readings.csv", monthly = TRUE)  
  
## End(Not run)
```



# Index

## \* datasets

fips\_lookup, 2

fips\_lookup, 2

hpms, 2, 4, 7

lottr, 3, 7

phed, 4, 7

score, 6

tpm, 7

tttr, 7, 7