

National Park Service



Calculation of Hydrologic Tools Indicator Metrics and Fundamental Daily Streamflow Statistics

DeVivo, J. C. 2018. Calculation of Hydrologic Indicator Tools metrics and Fundamental Daily Streamflow Statistics. Inventory & Monitoring Division Standard Operating Procedure NPS/IMD/SOP—3.3.1. National Park Service, Fort Collins, Colorado.

Each standard operating procedure (SOP) should have a summary that describes its content.

[illegible]

Contents

	Page
Summary	1
Revision Log.....	1
Overview.....	3
Preparation	3
Review of Monitoring Stations of Interest.....	4
Annual Calculation of Metrics.....	5
Quality Control	7
Identification of Errors Returned by R Scripts When Calculating Metrics.....	7
Validation of Metric Calculations	8
Period of Record Check.....	8
Data Suitability Check.....	8
Data Qualification Codes	9
Data Quality Evaluation	9
Literature Cited	10
Appendix A. RMD File Output for HIT Metric Calculations.....	11
Appendix V. RMD File Output for FDSS Metric Calculations.....	18

Overview

Calculation of metrics generally follows three steps. This SOP covers the calculation of metrics and QA/QC of resultant data. Note that only some procedures QA/QC procedures have been developed at the time of protocol publication.

1. Reviewing and updating stations of interest and periods of record of interest (see SOP 3.3.3; DeVivo 2018).
2. Calculating metrics for each period of record
 - a. Storing metric data in protocol database where calculations succeed
 - b. Storing null data and qualification flag for instances where calculations fail
3. Quality Assurance and Control
 - a. For periods of record where calculations succeed, evaluation of whether analysis code reproduces published results (from USGS) and flagging of data incorrectly reproduced. Data will be flagged by metric across all stations and periods of record.
 - b. For periods of record where calculations fail, evaluation of the period of record and one or more of the following:
 - i. Truncating of the period of record to remove data gaps at the beginning or end. The period of record is qualified as differing from the published period of record with an explanation.
 - ii. Splitting of the period of record to periods before and after discontinuities.
 - iii. Estimation or interpolation of missing data. Metrics calculated using interpolated or estimated data are qualified as based on incomplete data records.
 - c. For each period of record, metrics are also qualified based on underlying data adequacy, which will be used to inform trend analyses. This will be based on a combination of the duration of the period of record, the type of calculations used in deriving metrics (seasonal averages vs. rolling averages), and the amount of estimated vs. measured data included in the analysis.

Preparation

Required software:

- R and R Studio. The procedures below (and scripts for calculating methods) are based on R version 3.5.1 and may need to be updated based on your local installation.

Required R Packages. Please install with all dependencies:

- IHA (<https://rdr.io/rforge/IHA/>). Calculates a subset of *Indicators of Hydrologic Alteration* metrics as defined by The Nature Conservancy (TNC 2009). The Nature Conservancy's web page at <http://conserveonline.org/workspaces/iha> describes the software and has illustrations of its use. The IHA R package does not currently implement the EFC component of the software.
- dataRetrieval (<https://github.com/USGS-R/dataRetrieval>). Collection of functions to help retrieve U.S. Geological Survey (USGS) and U.S. Environmental Protection Agency (EPA) water quality and hydrology data from web services. USGS web services are discovered from National Water Information System (NWIS) <<https://waterservices.usgs.gov/>> and <<https://waterdata.usgs.gov/nwis>>. Both EPA and USGS water quality data are obtained from the Water Quality Portal <<https://www.waterqualitydata.us/>>.
- EflowStats (<https://github.com/USGS-R/EflowStats>). EflowStats is a reimplement of the Hydrologic Index Tool (HIT; Henriksen et al, 2006) for calculating 171 hydrologic indices for stream classification analysis. EflowStats also calculates 7 additional statistics used for streamflow classification referred to as the "Magnificent Seven" (MAG7, Archfield et al., 2013). Unlike the original HIT, EflowStats has been redesigned to ingest general hydrologic timeseries data and is not restricted to data formats used by the USGS National Water Information System.
- RODB (C) (<https://cran.r-project.org/web/packages/RODBC/index.html>). An ODBC database interface.
- Devtools (<https://github.com/hadley/devtools>). Required to install Eflowstats.

Before you begin

- You will need to set up an ODBC connection to allow the RODB (C) R package to read and write data to the JCD_Hydro database on the INPNISCVDBNRSSD\IMDGIS SQL Server instance.

Review of Monitoring Stations of Interest

These procedures are designed to be run for all stations and periods of records of interest. Please refer to SOP 3.3.3 (DeVivo 2018) for procedures for adding stations and custom periods of record to be used for analysis.

At the inception of protocol implementation, all USGS and EPA monitoring stations inside and within 1 km miles of park boundaries were used for analysis. Prior to each year's run of statistics, networks should be asked to review the current list of stations to determine if any others should be added, and identify any dates/events be used for before-after comparisons (such as the construction or removal of a dam). Note that many of the metrics to be calculated require a continuous period of

record (no data gaps) so the period of record used for analysis may be a subset of the full period of deployment for some stations (see SOP 3.3.3 for discussion on derivation of periods of record for analysis).

Annual Calculation of Metrics

For each station and period of record of interest, 171 hydrologic measures are calculated using R scripts available at https://github.com/nationalparkservice/EnvironmentalSettings_Hydro (Table 1 and

Table 2).

Table 1. Scripts to calculate HIT and FDSS metrics.

Script Name	Description
HIT_Metric_Calculations.Rmd	Scripts to run all HIT metrics (see Table 2 below)
FDSS_Metric_Calculations.Rmd	Script to run all FDSS metrics (see Table 2 below)

Table 2. Surface-water dynamics measures and description of associated parameters. Numbers in parentheses indicate the number of metrics in each category. Specific metric descriptions are provided in Appendix C. [* = parameters that are provided from source data].

Measure	Parameters	Description (number of metrics)	Reference
* Discharge	Discharge	Daily average discharge.	USGS 2017
Hydrologic Index Tool Metrics	Event Duration	Indices describing the duration of high (24), and low (20) flow conditions.	Henriksen et al. (2006)
	Event Frequency	Indices describing the frequency of high (11) and low (3) flow conditions.	
	Flow Magnitude	Indices describing the magnitude of average (45), high (27), and low (22) flow conditions.	
	Flow Timing	Indices describing the timing of average (3), high (3), and low (4) flow conditions	
	Rates of Change	Indices (9) that describe the rate of change of flow conditions	
Fundamental Daily Streamflow Statistics	Seasonal Signal	Indices (7) to describe and classify streams based on daily stream flow signal, including mean, coefficient of variation, skewness, kurtosis, the autoregressive lag-one, correlation coefficient, amplitude, and phase	Archfield et al. (2014)
	Rate and frequency of water condition changes	Rise and fall rates, and the number of hydrologic reversals (3)	

Quality Control

Identification of Errors Returned by R Scripts When Calculating Metrics

If errors are encountered when calculating HIT and FDSS statistics that results in metrics *not* being calculated, the R script will apply a diagnostic code to the period of record analyzed (Table 2). The codes are intended to provide the protocol lead with diagnostic information to determine whether to analyze data using alternate options (changes to the period of record being analyzed, or non-default parameters for the station).

Note that the R package provided by USGS is still in development and is expected to improve over time. Currently the existing data record for several periods of record for many stations does not allow for metric calculation, and it's unclear if this is a function of the code provided by USGS or the underlying data quality. It's important to be aware of which modules have changed prior to calculating metrics each year. In some cases metrics that have been flagged in prior years may be updated and “unflagged.” Until the USGS-provided tools stabilize it is expected that several metrics—and stations and periods of record for which they're calculated—will need to be qualified due to calculation errors, which will be rectified over time. In the meantime, QC procedures will be designed to identify instances where errors occur and to flag them for future re-evaluation when appropriate. This will be done by comparing data returned using the scripts developed as a part of

this protocol to the data published/ disseminated by USGS at the following site, replacing the XXX with the station ID number:

<https://cida.usgs.gov/nwc/#!/streamflow-stats/gage/xxxx>

Validation of Metric Calculations

To be added in Version 2.0.

Period of Record Check

To be added in Version 2.0.

Data Suitability Check

To be added in Version 2.0.

Data Qualification Codes

Where appropriate qualify data as follows:

Table 3. Data qualification codes used when calculating Hydrologic Indicator Tools (HIT) and Fundamental Daily Streamflow Statistics (FDSS) metrics.

Code	Applies To	Description	Remedy
AZ	Period of Record	All daily flow values equal zero	Metrics not calculated
MD	Period of Record	Missing daily flow data	Evaluate period of record for discontinuities and truncate/split as appropriate.
FTN	Period of Record	Flood threshold could not be calculated and instead set to null (affects Duration H, Frequency H, and Timing A,H, & L Metrics)	None; informs data use.
NAs	Period of Record	Error thrown during HIT calculations due to NAs in log transformed data	Metrics not calculated
TNCF-2	Period of Record	"mimic-TNC" setting set to FALSE for IHA group 2 metrics	None; informs data use.
TNCF-3	Period of Record	"mimic-TNC" setting set to FALSE for IHA group 3 metrics	None; informs data use.
MVS	Metric	Metric Value Suspect. The calculated metric value does not match accepted/ published value	Recalculate metrics upon release of updated code; omit from trend analyses.
MPE	Measure	Metric Precision Error. The calculated value does not match the precision of	Values are flagged but not recalculated.
MCE	Metric	Metric Calculation Error.	Recalculate metrics upon release of updated code; omit from trend analyses.
MCS	Metric	Metric Calculation Suspect. Code used to reproduce published USGS calculation does not return accepted/ published value.	Recalculate metrics upon release of updated code; omit from trend analyses.

Data Quality Evaluation

The following is summarized and disseminated with data:

- the total number of stations for which metrics are calculated
- the number of station X periods of record for which HIT and FDSS statistics are calculated
- the number of station X periods of record for which metrics are successfully calculated
- MCS and MVS qualification rates

Literature Cited

Archfield, S. A., J. G. Kennen, D. M. Carlisle, and D. M. Wolock. 2014. An objective and parsimonious approach for classifying natural flow regimes at a continental scale. *River Research and Applications* 30:1166-1183.

DeVivo, J. C. 2018. Adding new stations and periods of record for use in calculation of hydrologic measures as a part of the IMD Environmental Settings Protocol. Inventory & Monitoring Division Standard Operating Procedure NPS/IMD/SOP—3.3.3. National Park Service, Fort Collins, Colorado.

Henriksen, J. A., J. Heasley, J. G. Kennen, and S. Niewsand. 2006. Users' manual for the hydroecological integrity assessment process software (including the New Jersey assessment tools). Open File Report 2006-1093. U.S. Geological Survey, Biological Resources Discipline, Reston, Virginia.

The Nature Conservancy (TNC). 2009. Indicators of Hydrologic Alteration, software version 7.1.0.10.

Appendix A. RMD File Output for HIT Metric Calculations

This vignette is current as of 15 November 2018. For the latest version, see:

https://github.com/joe-devivo/ESHydrology/blob/master/vignettes/HIT_Metric_Calculations.Rmd

Before you begin

You will need to set up an ODBC connection named “Hydro” to allow the RODBC R package to read and write data to the JCD_Hydro database on the INPNISCVDBNRSSDSQL Server instance.

You will also need the following R packages installed:

- IHA (<https://rdr.io/rforge/IHA/>). Calculates a subset of Indicators of Hydrologic Alteration metrics as defined by The Nature Conservancy (TNC 2009). The Nature Conservancy’s web page at <http://conserveonline.org/workspaces/ih> describes the software and has illustrations of it’s use. The IHA R package does not currently implement the EFC component of the software.
- dataRetrieval (<https://github.com/USGS-R/dataRetrieval>). Collection of functions to help retrieve U.S. Geological Survey (USGS) and U.S. Environmental Protection Agency (EPA) water quality and hydrology data from web services. USGS web services are discovered from National Water Information System (NWIS) <https://waterservices.usgs.gov/> and <https://waterdata.usgs.gov/nwis>. Both EPA and USGS water quality data are obtained from the Water Quality Portal <https://www.waterqualitydata.us/>.
- EflowStats (<https://github.com/USGS-R/EflowStats>). EflowStats is a reimplementaion of the Hydrologic Index Tool (HIT; Henriksen et al, 2006) for calculating 171 hydrologic indices for stream classification analysis. EflowStats also calculates 7 additional statistics used for streamflow classification referred to as the “Magnificent Seven” (MAG7, Archfield et al., 2013). Unlike the original HIT, EflowStats has been redesigned to ingest general hydrologic timeseries data and is not restricted to data formats used by the USGS National Water Information System.
- RODBC (<https://cran.r-project.org/web/packages/RODBC/index.html>). An ODBC database interface.
- Devtools (<https://github.com/hadley/devtools>). Required to install Eflowstats.

Calculation of HIT and FDSS Metrics

Pull station data from master database

The following pulls data from the master hydrology database

```
ch <- odbcConnect("Hydro")
POR2run<-sqlFetch(ch,"dbo.vw_site_no_X_PORs_for_Analysis",as.is = c(TRUE), stringsAsFactors = FALSE)
siteInfo<-sqlFetch(ch,"dbo.tbl_siteInfo",as.is = c(TRUE), stringsAsFactors = FALSE)
MasterPORs<-sqlFetch(ch,"dbo.tlu_PORs",as.is = c(TRUE), stringsAsFactors = FALSE)
odbcClose(ch)
```

```

colnames(MasterPORs)[5] <- "AnalysisStartDate"
colnames(MasterPORs)[6] <- "AnalysisEndDate"

POR2run<-merge(POR2run,MasterPORs,by="POR_ID")
POR2run<-merge(POR2run,siteInfo,by="siteInfoID")

POR2run$AnalysisStartDate<-ymd(as.Date(POR2run$AnalysisStartDate))
POR2run$AnalysisEndDate<-ymd(as.Date(POR2run$AnalysisEndDate))

```

Run IHA calculations

The following code runs a loop X number of times based on parameters set in the header of this Rmd file. By default, the number of loops is set to 1, but this can be easily changed by “knitting with parameters.”

Note that this code chunk will only work if the POR2run table contains a period of record / station combination in need of processing.

```

totalchunks<-params$loops2run
rowstowrite<-params$rowsperloop
currentchunk<-1

# Start Loop
repeat
{

ch <-  odbcConnect("Hydro")
RunHIT<-sqlFetch(ch,"dbo.vw_HIT_DurationHigh_by_AnalysisPORID",as.is = c(TRUE
), stringsAsFactors = FALSE)
odbcClose(ch)

POR2run<-anti_join(POR2run,RunHIT,by="AnalysisPORID")

rm(MAwrite)
rm(MLwrite)
rm(MHwrite)
rm(FLwrite)
rm(FHwrite)
rm(DLwrite)
rm(DHwrite)
rm(TAwrite)
rm(TLwrite)
rm(THwrite)
rm(RAwrite)
totalrows<-nrow(POR2run)
currentrow<-1

repeat

```

```

{
  sn<-POR2run[currentrow,11]
  sd<-POR2run[currentrow,8]
  ed<-POR2run[currentrow,9]
  AnalysisPORID<-POR2run[currentrow,3]
  DrainArea<-POR2run[currentrow,39]

  dailyQ <- readNWISdv(siteNumber = sn, parameterCd = "00060", startDate = sd,
endDate = ed)

  dailyQClean <- validate_data(dailyQ[c("Date","X_00060_00003")],yearType="water")

  # calculate flood threshold for POR
  peakFlows <-readNWISpeak(siteNumbers=sn,startDate=sd,endDate=ed)
  peakFlowMissingData=any(is.na(peakFlows$peak_va))
  peakFlowMissingDate=any(is.na(peakFlows$peak_dt))
  peakFlowErrorCode=any(!is.na(peakFlows$peak_cd))

  #Get flood recurrence threshold

  if (nrow(peakFlows)==1) {
    floodThresh=NULL
  } else if (peakFlowMissingData==TRUE) {
    floodThresh=NULL
  } else if (peakFlowMissingDate==TRUE){
    floodThresh=NULL
  } else if (peakFlowErrorCode==TRUE){
    floodThresh=NULL
  } else if (nrow(peakFlows)==0){
    floodThresh=NULL
  } else {
    floodThresh <- get_peakThreshold(dailyQClean[c("date","discharge")], peak
Flows[c("peak_dt","peak_va")], yearType = "water")
  }

  if (is.null(floodThresh)) {
    ft=NA
    PORQualOut<-data.frame(cbind(PORQualifierID=4,AnalysisPORID=POR2run[curre
ntrow,3]))
    ch <- odbcConnect("Hydro")
    sqlSave(ch,PORQualOut,"dbo.vw_QualifyPORs",append=TRUE,rownames=FALSE)
    odbcClose(ch)
  }

  test<-try(calc_allHIT(dailyQClean, yearType = "water", stats = "all", digits
= 3, pref = "mean", drainArea = as.numeric(DrainArea), floodThreshold = flood
Thresh))

```

```

if (class(test)=="try-error"){

  PORQualOut<-data.frame(cbind(PORQualifierID=5,AnalysisPORID=POR2run[curre
ntrow,3]))
  ch <- odbcConnect("Hydro")
  sqlSave(ch,PORQualOut,"dbo.vw_QualifyPORs",append=TRUE,rownames=FALSE)
  odbcClose(ch)

} else {

output<-calc_allHIT(dailyQClean, yearType = "water", stats = "all", digits =
3, pref = "mean", drainArea = as.numeric(DrainArea), floodThreshold = floodTh
resh)
output.with.rownames <- data.frame(output[,-1], row.names=output[,1])

is.na(output.with.rownames)<-sapply(output.with.rownames, is.infinite)
output.with.rownames[is.na(output.with.rownames)]<-NA

output2<-data.frame(t(output.with.rownames))

#add POR, Site number, and QA codes to data frame and write to database

MA<-cbind(AnalysisPORID=AnalysisPORID,output2[,1:45], runtime=Sys.time())
ML<-cbind(AnalysisPORID=AnalysisPORID,output2[,46:67], runtime=Sys.time())
MH<-cbind(AnalysisPORID=AnalysisPORID,output2[,68:94], runtime=Sys.time())
FL<-cbind(AnalysisPORID=AnalysisPORID,output2[,95:97], runtime=Sys.time())
FH<-cbind(AnalysisPORID=AnalysisPORID,output2[,98:108], runtime=Sys.time())
DL<-cbind(AnalysisPORID=AnalysisPORID,output2[,109:128], runtime=Sys.time())

if (is.null(floodThresh)) {
  DH<-cbind(AnalysisPORID=AnalysisPORID,floodthreshold=NA,output2[,129:152]
, runtime=Sys.time())
} else {
  DH<-cbind(AnalysisPORID=AnalysisPORID,floodthreshold=floodThresh,output2[
,129:152], runtime=Sys.time())
}

TA<-cbind(AnalysisPORID=AnalysisPORID,output2[,153:155], runtime=Sys.time())
TL<-cbind(AnalysisPORID=AnalysisPORID,output2[,156:159], runtime=Sys.time())
TH<-cbind(AnalysisPORID=AnalysisPORID,output2[,160:162], runtime=Sys.time())
RA<-cbind(AnalysisPORID=AnalysisPORID,output2[,163:171], runtime=Sys.time())

if (exists('MAwrite')) {
  MAwrite<-rbind(MAwrite,MA)
} else {
  MAwrite=MA
}
}

```

```

if (exists('MLwrite')) {
  MLwrite<-rbind(MLwrite,ML)
} else {
  MLwrite=ML
}

if (exists('MHwrite')) {
  MHwrite<-rbind(MHwrite,MH)
} else {
  MHwrite=MH
}

if (exists('FLwrite')) {
  FLwrite<-rbind(FLwrite,FL)
} else {
  FLwrite=FL
}

if (exists('FHwrite')) {
  FHwrite<-rbind(FHwrite,FH)
} else {
  FHwrite=FL
}

if (exists('DLwrite')) {
  DLwrite<-rbind(DLwrite,DL)
} else {
  DLwrite=DL
}

if (exists('DHwrite')) {
  DHwrite<-rbind(DHwrite,DH)
} else {
  DHwrite=DH
}

if (exists('TAwrite')) {
  TAwrite<-rbind(TAwrite,TA)
} else {
  TAwrite=TA
}

if (exists('TLwrite')) {
  TLwrite<-rbind(TLwrite,TL)
} else {
  TLwrite=TL
}

if (exists('THwrite')) {
  THwrite<-rbind(THwrite,TH)
}

```

```

    } else {
      THwrite=TH
    }

if (exists('RAwrite')) {
  RAwrite<-rbind(RAwrite,RA)
} else {
  RAwrite=RA
}

}

# End Loop

print(c("chunk",totalchunks-currentchunk+1,"row",rowstowrite-currentrow+1))
currentrow = currentrow+1

if(currentrow==rowstowrite+1)
{
  break
}
}

ch <- odbcConnect("Hydro")
sqlSave(ch,MAwrite,"dbo.vw_HIT_MagAverage_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,MLwrite,"dbo.vw_HIT_MagLow_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,MHwrite,"dbo.vw_HIT_MagHigh_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,FLwrite,"dbo.vw_HIT_FrequencyLow_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,FHwrite,"dbo.vw_HIT_FrequencyHigh_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,DLwrite,"dbo.vw_HIT_DurationLow_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,DHwrite,"dbo.vw_HIT_DurationHigh_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,TAwrite,"dbo.vw_HIT_TimingAverage_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,TLwrite,"dbo.vw_HIT_TimingLow_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,THwrite,"dbo.vw_HIT_TimingHigh_by_AnalysisPORID",append=TRUE,rownames=FALSE)
sqlSave(ch,RAwrite,"dbo.vw_HIT_RateChange_by_AnalysisPORID",append=TRUE,rownames=FALSE)

odbcClose(ch)

# End Loop

```

```

print(c("chunk written"))
currentchunk = currentchunk+1
if(currentchunk==totalchunks+1)
{
    break
}
}

## Error in inherits(x, c("xml_node", "xml_document", "xml_nodeset")): HTTP S
tatus 400 - Site number length must be no less than 8 numeric digits at argum
ent 0

```

Quality Assurance and Quality Control Procedures

Validation of Metric Calculations

To be added in Version 2.0

Period of Record Check

To be added in Version 2.0

Data Suitability Check

To be added in Version 2.0

Data Quality Evaluation

The following is summarized and disseminated with data:

- the total number of stations for which metrics are calculated
- the number of station X periods of record for which IHA statistics are calculated
- the number of station X periods of record for which metrics are successfully calculated
- Qualification rates

Methods for doing so will be included in Version 2.0.

Appendix V. RMD File Output for FDSS Metric Calculations

This vignette is current as of 15 November 2018. For the latest version, see:

https://github.com/joe-devivo/ESHydrology/blob/master/vignettes/FDSS_Metric_Calculations.Rmd

Before you begin

You will need to set up an ODBC connection named “Hydro” to allow the RODBC R package to read and write data to the JCD_Hydro database on the INPNISCVDBNRSSDSQL Server instance.

You will also need the following R packages installed:

- IHA (<https://rdr.io/rforge/IHA/>). Calculates a subset of Indicators of Hydrologic Alteration metrics as defined by The Nature Conservancy (TNC 2009). The Nature Conservancy’s web page at <http://conserveonline.org/workspaces/ih> describes the software and has illustrations of it’s use. The IHA R package does not currently implement the EFC component of the software.
- dataRetrieval (<https://github.com/USGS-R/dataRetrieval>). Collection of functions to help retrieve U.S. Geological Survey (USGS) and U.S. Environmental Protection Agency (EPA) water quality and hydrology data from web services. USGS web services are discovered from National Water Information System (NWIS) <https://waterservices.usgs.gov/> and <https://waterdata.usgs.gov/nwis>. Both EPA and USGS water quality data are obtained from the Water Quality Portal <https://www.waterqualitydata.us/>.
- EflowStats (<https://github.com/USGS-R/EflowStats>). EflowStats is a reimplementaion of the Hydrologic Index Tool (HIT; Henriksen et al, 2006) for calculating 171 hydrologic indices for stream classification analysis. EflowStats also calculates 7 additional statistics used for streamflow classification referred to as the “Magnificent Seven” (MAG7, Archfield et al., 2013). Unlike the original HIT, EflowStats has been redesigned to ingest general hydrologic timeseries data and is not restricted to data formats used by the USGS National Water Information System.
- RODBC (<https://cran.r-project.org/web/packages/RODBC/index.html>). An ODBC database interface.
- Devtools (<https://github.com/hadley/devtools>). Required to install Eflowstats.

Calculation of HIT and FDSS Metrics

Pull station data from master database

The following pulls data from the master hydrology database

```
ch <- odbcConnect("Hydro")
POR2run<-sqlFetch(ch,"dbo.vw_site_no_X_PORs_for_Analysis",as.is = c(TRUE), stringsAsFactors = FALSE)
siteInfo<-sqlFetch(ch,"dbo.tbl_siteInfo",as.is = c(TRUE), stringsAsFactors = FALSE)
MasterPORs<-sqlFetch(ch,"dbo.tlu_PORs",as.is = c(TRUE), stringsAsFactors = FALSE)
odbcClose(ch)
```

```

colnames(MasterPORs)[5] <- "AnalysisStartDate"
colnames(MasterPORs)[6] <- "AnalysisEndDate"

POR2run<-merge(POR2run,MasterPORs,by="POR_ID")
POR2run<-merge(POR2run,siteInfo,by="siteInfoID")

POR2run$AnalysisStartDate<-ymd(as.Date(POR2run$AnalysisStartDate))
POR2run$AnalysisEndDate<-ymd(as.Date(POR2run$AnalysisEndDate))

```

Run IHA calculations

The following code runs a loop X number of times based on parameters set in the header of this Rmd file. By default, the number of loops is set to 1, but this can be easily changed by “knitting with parameters.”

Note that this code chunk will only work if the POR2run table contains a period of record / station combination in need of processing.

```

totalchunks<-params$loops2run
rowstowrite<-params$rowsperloop
currentchunk<-1

# Start Loop
repeat
{

ch <- odbcConnect("Hydro")
RunFDSS<-sqlFetch(ch,"dbo.vw_FDSS_by_AnalysisPORID",as.is = c(TRUE), stringsAsFactors = FALSE)
odbcClose(ch)

POR2run<-anti_join(POR2run,RunFDSS,by="AnalysisPORID")

rm(magnifStatsOut3)
totalrows<-nrow(POR2run)
currentrow<-1

repeat
{

sn<-POR2run[currentrow,11]
sd<-POR2run[currentrow,8]
ed<-POR2run[currentrow,9]
AnalysisPORID<-POR2run[currentrow,3]

dailyQ <- readNWISdv(siteNumber = sn, parameterCd = "00060", startDate = sd,

```

```

endDate = ed)

dailyQClean <- validate_data(dailyQ[c("Date", "X_00060_00003")], yearType="water")

if (max(dailyQClean$discharge)) == 0 {

PORQualOut<-cbind(PORQualifierID=1, AnalysisPORID=POR2run[currentrow,3])
ch <- odbcConnect("Hydro")
sqlSave(ch, PORQualOut, "dbo.vw_QualifyPORs", append=TRUE, rownames=FALSE)
odbcClose(ch)

} else {

magnifStatsOut <- calc_magnifSeven(dailyQClean, yearType="water", digits=3)
magnifStatsOut.with.rownames <- data.frame(magnifStatsOut[, -1], row.names=magnifStatsOut[, 1])
magnifStatsOut2<-t(magnifStatsOut.with.rownames)

#add POR, Site number, and QA codes to data frame and write to database
magnifStatsOut2<-cbind(AnalysisPORID=AnalysisPORID, magnifStatsOut2, data.frame(site_no=sn, runtime=Sys.time()))

if (exists('magnifStatsOut3')) {
  magnifStatsOut3<-rbind(magnifStatsOut3, magnifStatsOut2)
} else {
  magnifStatsOut3=magnifStatsOut2
}
}
# End Loop
print(c("chunk", totalchunks-currentchunk+1, "row", rowstowrite-currentrow+1))
currentrow = currentrow+1
if(currentrow==rowstowrite+1)
{
  break
}
}

ch <- odbcConnect("Hydro")
sqlSave(ch, magnifStatsOut3, "dbo.vw_FDSS_by_AnalysisPORID", append=TRUE, rownames=FALSE)
odbcClose(ch)

# End Loop
print(c("chunk written"))
currentchunk = currentchunk+1
if(currentchunk==totalchunks+1)
{
  break
}

```

```
}  
}  
## Error: <text>:34:33: unexpected '=='  
## 33:  
## 34: if (max(dailyQClean$discharge)) ==  
##                                     ^
```

Quality Assurance and Quality Control Procedures

Validation of Metric Calculations

To be added in Version 2.0

Period of Record Check

To be added in Version 2.0

Data Suitability Check

To be added in Version 2.0

Data Quality Evaluation

The following is summarized and disseminated with data:

- the total number of stations for which metrics are calculated
- the number of station X periods of record for which IHA statistics are calculated
- the number of station X periods of record for which metrics are successfully calculated
- Qualification rates

Methods for doing so will be included in Version 2.0.