

Introduction

The Flemish vegetation database, INBOVEG, is an application developed to provide a repository of relevés and makes the relevés available for future use.

INBOVEG supports different types of recordings: BioHab recordings (protocol of Natura 2000 monitoring) and the classic relevés. The classic relevés can stand alone, be an element of a collection or element of a chain where the linkage is used to give information about the relative position of recording within a series. Ample selection and export functions toward analysis tools are provided. It also provides standardized lists of species, habitats, life forms, scales etc. Original observations are preserved and a full history of subsequent identifications is saved.

Aim

In this tutorial we make functions available to query data directly from the INBOVEG SQL-server database. This to avoid writing your own queries or to copy/paste them from the access-frontend for INBOVEG.

We have provided functions to query * survey (INBOVEG-projects) * recordings (vegetation relevés) * metadata of recordings (header info) * classification (Natura2000 or local classification like BWK) * qualifiers (management and site characteristics)

Packages and connection

In order to run the functionalities, some R packages need to be installed.

The following packages are needed to run this code: * glue * DBI * assertthat * dplyr

Loading the functionality can be done by loading the `inborutils` package: * inborutils

Be sure you have reading-rights for CYDONIA otherwise place an ICT-call (<ict.helpdesk@inbo.be>)

```
library(glue)
library(DBI)
library(assertthat)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following object is masked from 'package:glue':
##
##      collapse

## The following objects are masked from 'package:stats':
##
##      filter, lag

## The following objects are masked from 'package:base':
##
##      intersect, setdiff, setequal, union

library(knitr)
library(inborutils)

## Registered S3 methods overwritten by 'ggplot2':
##   method      from
##   [.quosures  rlang
##   c.quosures  rlang
##   print.quosures rlang

opts_chunk$set(echo = TRUE)
```

The following R-code can be used to establish a connection to INBOVEG by means of a connection string:

```
<!--better to use a connection string than dsn.
dsn requires extra steps and settings in windows odbc manager-->
connection <- dbConnect(odbc::odbc(), .connection_string = "Driver=SQL Server;Server=inbo-sql07-
prd.inbo.be,1433;Database=D0010_00_Cydonia;Trusted_Connection=Yes;")
```

Or using dbconnection of the inborutils-package with the database 'Cydonia' on the inbo-sql07-prd server:

```

con <- connect_inbo_dbase("D0010_00_Cydonia")

Functionality
-----

### Survey information

The function 'inboveg\_survey' queries the INBOVEG database for survey
information (metadata about surveys) for one or more survey(s) by the
name of the survey.

#### Examples

get information of a specific survey and collect data

  survey_info <- inboveg_survey(con, survey_name = "OudeLanden_1979", collect = TRUE)

get information of all surveys and collect data

  allsurveys <- inboveg_survey(con)

only a part of the survey name is known?

  partsurveys <- inboveg_survey(con, survey = "%MILKLIM%")

### Recording information

The function 'inboveg\_recordings' queries the INBOVEG database for
relevé information (which species were recorded in which plots and in
which vegetation layers with which cover) for one or more surveys.

#### Examples

get the relevés from one survey and collect the data

  recording_heischraal2012 <- inboveg_recordings(con, survey_name =
"MILKLIM_Heischraal2012", collect = TRUE)

get all recordings from MILKLIM surveys (partial matching), don't
collect

  recording_milkim <- inboveg_recordings(con, survey_name = "%MILKLIM%",
collect = TRUE)

get recordings from several specific surveys

  recording_severalsurveys <- inboveg_recordings(con, survey_name =
c("MILKLIM_Heischraal2012", "NICHE Vlaanderen"), multiple = TRUE,
collect = TRUE)

get all relevés of all surveys, don't collect the data

  allrecordings <- inboveg_recordings(con)

### Header information

This function queries the INBOVEG database for header information
(metadata for a vegetation-relevé) for one survey by the name of the
survey and the recorder type.

#### Examples

get header information from a specific survey and a specific recording
type and collect the data

  header_info <- inboveg_header(con, survey_name = "OudeLanden_1979",
rec_type = "Classic", collect = TRUE)

get header information of all surveys, don't collect the data

  all_header_info <- inboveg_header(con)

### Classification information

The function 'inboveg\_classification' queries the INBOVEG database for
information on the field classification (N2000 or BWK-code) of the
relevé for one or more survey(s) by the name of the survey.

#### Examples

get a specific classification from a survey and collect the data

  classif_info <- inboveg_classification(con,
survey_name = "MILKLIM_Heischraal2012", classif = "4010", collect = TRUE)

```

get all surveys, all classifications, don't collect the data

```
alleccodes <- inboveg_classification(con)
```

Qualifiers information

Nog uitwerken, eerst functie in orde krijgen

More complex queries

These functions gives the basis information out of INBOVEG.

```
<!-- ### hieronder de oude versie -->
<!-- # Retrieving data -->
<!-- ## *iv_Survey*: -->
<!-- gives the list of all surveys in InboVeg -->
<!-- - define the name of the survey by survey <- "name" -->
<!-- ``{r} -->
<!-- survey_info <- function(survey, con) { -->
<!--   dbGetQuery(con, glue_sql( -->
<!--     "SELECT -->
<!--       ivS.Id -->
<!--       , ivS.Name -->
<!--       , ivS.Description -->
<!--       , ivS.Owner -->
<!--       , ivS.creator -->
<!--     FROM [dbo].[ivSurvey] ivS -->
<!--     WHERE ivS.Name LIKE {survey}", -->
<!--       ivS.Name = survey, -->
<!--       .con = con )) -->
<!-- } -->
<!-- `` -->
<!-- Example -->
<!-- * survey <- "OudeLanden_1979" -->
<!-- * SurveyInfo <- survey_info(survey, con) -->
<!-- * SurveyInfo -->
<!-- The whole list of surveys is given by -->
<!-- *AllSurveys <- survey_info(survey = "%", .con = con) -->
<!-- Only a part of the survey name is known? -->
<!-- *PartSurveys <- survey_info(survey = "%MILKLIM%", .con = con) -->
<!-- ## *iv_headerinfo*: -->
<!-- gives the metadata for a vegetation-relevé (one row per vegetation-relevé identified by 'RecordingGivid') -->
<!-- - specify two parameters for the function: -->
<!--   - RecType = c('Classic', 'Classic-emmer', 'Classic-ketting', 'BioHab', 'ABS') -->
<!--   - SurveyName = to get the list, run the code under "## iv_survey -->
<!-- ``{r} -->
<!-- header_info <- function(SurveyName, RecType, .con) { -->
<!--   dbGetQuery(con, glue_sql( -->
<!--     "SELECT -->
<!--       ivR.[RecordingGivid] -->
<!--       , ivS.Name -->
<!--       , ivR.UserReference -->
<!--       , ivR.LocationCode -->
<!--       , ivR.Latitude -->
<!--       , ivR.Longitude -->
<!--       , ivR.Area -->
<!--       , ivR.Length -->
<!--       , ivR.Width -->
<!--       , ivR.SurveyId -->
<!--       , ivR.RecTypeID -->
<!--       , coalesce(area, convert( nvarchar(20),ivR.Length * ivR.Width)) as B -->
<!--     FROM [dbo].[ivRecording] ivR -->
<!--     INNER JOIN [dbo].[ivSurvey] ivS on ivS.Id = ivR.SurveyId -->
<!--     INNER JOIN [dbo].[ivRecTypeD] ivRec on ivRec.ID = ivR.RecTypeID -->
<!--     where ivR.NeedsWork = 0 -->
<!--     AND ivS.Name LIKE {SurveyName} -->
<!--     AND ivRec.Name LIKE {RecType}", -->
<!--       ivS.Name = SurveyName, -->
<!--       ivRec.Name = RecType, -->
<!--       .con = con)) -->
<!-- } -->
<!-- `` -->
<!-- Example -->
<!-- * RecType <- "Classic" -->
<!-- * SurveyName <- "OudeLanden_1979" -->
<!-- * Headerinfo <- header_info(SurveyName, RecType, con) -->
<!-- * Headerinfo <- header_info("OudeLanden_1979", "Classic", con) -->
<!-- ## *iv_Classification_N2000*: -->
<!-- gives the N2000-code, recorderd by the observer of the relevé at the field (with or without field-key) -->
<!-- - specify the name of the survey you want to use. if none, all the classification records in inboveg will be
given -->
<!-- - specify the N2000 code to retrieve all relevés indicated as this code -->
<!-- ``{r} -->
```

```

<!-- classification_info_N2000 <- function(SurveyName, N2000, .con) { -->
<!-- dbGetQuery(con, glue_sql( -->
<!-- "SELECT -->
<!-- ivR.RecordingGivid -->
<!-- , ivS.Name as survey -->
<!-- , ivRLClas.Classif -->
<!-- , ivRLRes_Class.ActionGroup -->
<!-- , ivRLRes_Class.ListName -->
<!-- , ftN2k.Description as Habitattype -->
<!-- , ivRLClas.Cover -->
<!-- , ftC.PctValue -->
<!-- FROM ivRecording ivR -->
<!-- INNER JOIN ivSurvey ivS on ivS.Id = ivR.surveyId -->
<!-- LEFT JOIN [dbo].[ivRLClassification] ivRLClas on ivRLClas.RecordingID = ivR.Id -->
<!-- LEFT JOIN [dbo].[ivRLResources] ivRLRes_Class on ivRLRes_Class.ResourceGIVID = ivRLClas.ClassifResource -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftActionGroupList] ftAGL_Class on ftAGL_Class.ActionGroup =
ivRLRes_Class.ActionGroup collate Latin1_General_CI_AI -->
<!-- AND ftAGL_Class.ListName = ivRLRes_Class.ListName collate Latin1_General_CI_AI -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftN2kValues] ftN2K on ftN2K.Code = ivRLClas.Classif collate Latin1_General_CI_AI
-->
<!-- AND ftN2K.ListGIVID = ftAGL_Class.ListGIVID -->
<!-- LEFT JOIN [dbo].[ivRLResources] ivRLR_C on ivRLR_C.ResourceGIVID = ivRLClas.CoverResource -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftActionGroupList] ftAGL_C on ftAGL_C.ActionGroup = ivRLR_C.ActionGroup collate
Latin1_General_CI_AI -->
<!-- AND ftAGL_C.ListName = ivRLR_C.ListName collate Latin1_General_CI_AI -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftCoverValues] ftC on ftC.Code = ivRLClas.Cover collate Latin1_General_CI_AI -->
<!-- AND ftAGL_C.ListGIVID = ftC.ListGIVID -->
<!-- WHERE ivRLClas.Classif is not NULL -->
<!-- AND ivS.Name LIKE {SurveyName} -->
<!-- AND ivRLClas.Classif LIKE {N2000}", -->
<!-- ivS.Name = SurveyName, -->
<!-- ivRLClas.Classif = N2000, -->
<!-- .con = con)) -->
<!-- } -->
<!-- `` -->
<!-- Example -->
<!-- * SurveyName <- "MILKLIM_Heischraal2012" -->
<!-- * N2000 <- "4010" -->
<!-- * Classification <- classification_info_N2000(SurveyName, N2000, con) -->
<!-- * Classification2 <- classification_info_N2000("MILKLIM_Heischraal2012", "4010", con) -->
<!-- ## iv_Classification_BWK -->
<!-- gives the BWK-code, recorderd by the observer of the relev  -->
<!-- - specify the name of the survey you want to use. if none, all the classification records in inboveg will be
given -->
<!-- - specify the bwk-code to retrieve all relev s indicated as this code -->
<!-- ``{r} -->
<!-- classification_info_bwk <- function(SurveyName, BWK, .con) { -->
<!-- dbGetQuery(con, glue_sql( -->
<!-- "SELECT -->
<!-- ivR.RecordingGivid -->
<!-- , ivS.Name as survey -->
<!-- , ivRLClas.Classif -->
<!-- , ivRLRes_Class.ActionGroup -->
<!-- , ivRLRes_Class.ListName -->
<!-- , ftBWK.Description as LocalClassification -->
<!-- , ivRLClas.Cover -->
<!-- , ftC.PctValue -->
<!-- FROM ivRecording ivR -->
<!-- INNER JOIN ivSurvey ivS on ivS.Id = ivR.surveyId -->
<!-- LEFT JOIN [dbo].[ivRLClassification] ivRLClas on ivRLClas.RecordingID = ivR.Id -->
<!-- LEFT JOIN [dbo].[ivRLResources] ivRLRes_Class on ivRLRes_Class.ResourceGIVID = ivRLClas.ClassifResource -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftActionGroupList] ftAGL_Class on ftAGL_Class.ActionGroup =
ivRLRes_Class.ActionGroup collate Latin1_General_CI_AI -->
<!-- AND ftAGL_Class.ListName = ivRLRes_Class.ListName collate Latin1_General_CI_AI -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftBWKValues] ftBWK on ftBWK.Code = ivRLClas.Classif collate Latin1_General_CI_AI
-->
<!-- AND ftBWK.ListGIVID = ftAGL_Class.ListGIVID -->
<!-- LEFT JOIN [dbo].[ivRLResources] ivRLR_C on ivRLR_C.ResourceGIVID = ivRLClas.CoverResource -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftActionGroupList] ftAGL_C on ftAGL_C.ActionGroup = ivRLR_C.ActionGroup collate
Latin1_General_CI_AI -->
<!-- AND ftAGL_C.ListName = ivRLR_C.ListName collate Latin1_General_CI_AI -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftCoverValues] ftC on ftC.Code = ivRLClas.Cover collate Latin1_General_CI_AI -->
<!-- AND ftAGL_C.ListGIVID = ftC.ListGIVID -->
<!-- WHERE ivRLClas.Classif is not NULL -->
<!-- AND ivS.Name LIKE {SurveyName} -->
<!-- AND ivRLClas.Classif LIKE {BWK}", -->
<!-- ivS.Name = SurveyName, -->
<!-- ivRLClas.Classif = BWK, -->
<!-- .con = con)) -->
<!-- } -->
<!-- `` -->
<!-- Example -->
<!-- * SurveyName <- "CultuurgraslandTypologie" -->
<!-- * BWK <- "h" -->
<!-- * Classification <- classification_info_bwk(SurveyName, BWK, con) -->

```

```

<!-- * Classification2 <- classification_info_bwk("CultuurgraslandTypologie", "h", con) -->
<!-- ## iv_classification -->
<!-- gives the N2000-code or BWK-code, recorderd by the observer of the relev  at the field (with or without field-
key) -->
<!-- - specify the name of the survey you want to use. if none, all the classification records in inboveg will be
given -->
<!-- - specify the N2000 or BWK code to retrieve all relev s indicated as this code -->
<!-- ``{r} -->
<!-- classification_info_alles <- function(SurveyName, Classif, .con) { -->
<!-- dbGetQuery(con, glue_sql( -->
<!-- "Select ivR.RecordingGivid -->
<!-- , ivS.Name -->
<!-- , ivRLClas.Classif -->
<!-- , ivRLRes_Class.ActionGroup -->
<!-- , ivRLRes_Class.ListName -->
<!-- , ftBWK.Description as LocalClassification -->
<!-- , ftN2k.Description as Habitatttype -->
<!-- , ivRLClas.Cover -->
<!-- , ftC.PctValue -->
<!-- FROM ivRecording ivR -->
<!-- INNER JOIN ivSurvey ivS on ivS.Id = ivR.surveyId -->
<!-- LEFT JOIN [dbo].[ivRLClassification] ivRLClas on ivRLClas.RecordingID = ivR.Id -->
<!-- LEFT JOIN [dbo].[ivRLResources] ivRLRes_Class on ivRLRes_Class.ResourceGIVID = ivRLClas.ClassifResource -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftActionGroupList] ftAGL_Class on ftAGL_Class.ActionGroup =
ivRLRes_Class.ActionGroup collate Latin1_General_CI_AI -->
<!-- AND ftAGL_Class.ListName = ivRLRes_Class.ListName collate Latin1_General_CI_AI -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftBWKValues] ftBWK on ftBWK.Code = ivRLClas.Classif collate Latin1_General_CI_AI
-->
<!-- AND ftBWK.ListGIVID = ftAGL_Class.ListGIVID -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftN2kValues] ftN2K on ftN2K.Code = ivRLClas.Classif collate Latin1_General_CI_AI
-->
<!-- AND ftN2K.ListGIVID = ftAGL_Class.ListGIVID -->
<!-- LEFT JOIN [dbo].[ivRLResources] ivRLR_C on ivRLR_C.ResourceGIVID = ivRLClas.CoverResource -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftActionGroupList] ftAGL_C on ftAGL_C.ActionGroup = ivRLR_C.ActionGroup collate
Latin1_General_CI_AI -->
<!-- AND ftAGL_C.ListName = ivRLR_C.ListName collate Latin1_General_CI_AI -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftCoverValues] ftC on ftC.Code = ivRLClas.Cover collate Latin1_General_CI_AI -->
<!-- AND ftAGL_C.ListGIVID = ftC.ListGIVID -->
<!-- WHERE ivRLClas.Classif is not NULL ", -->
<!-- ivS.Name = SurveyName, -->
<!-- ivRLClas.Classif = Classif, -->
<!-- .con = con)) -->
<!-- } -->
<!-- `` -->
<!-- Example -->
<!-- *SurveyName <- "MILKLIM_Heischraal2012" -->
<!-- *Classif <- "4010" -->
<!-- *Classif_info <- classification_info_alles(SurveyName, Classif, con) -->
<!-- *Classif_info2 <- classification_info_alles("MILKLIM_Heischraal2012", "4010", con) -->
<!-- *Allecodes <- classification_info_alles(SurveyName = "%", Classif = "%", .con = con) -->
<!-- ## iv_Relev s -->
<!-- gives the relev s (plant list with coverage) of one Survey -->
<!-- - specify the name of the survey you want to use. if none, all the records in inboveg will be given (to avoid!)
-->
<!-- ``{r} -->
<!-- relev _info_surveyname <- function(SurveyName, .con) { -->
<!-- dbGetQuery(con, glue_sql( -->
<!-- "SELECT ivS.Name -->
<!-- , ivR.[RecordingGivid] -->
<!-- , ivRL_Layer.LayerCode -->
<!-- , ivRL_Layer.CoverCode -->
<!-- , ivRL_Iden.TaxonFullText as OriginalName -->
<!-- , Synoniem.ScientificName -->
<!-- , ivRL_Iden.PhenologyCode -->
<!-- , ivRL_Taxon.CoverageCode -->
<!-- , ftCover.PctValue -->
<!-- , ftAGL.Description as RecordingScale -->
<!-- FROM dbo.ivSurvey ivS -->
<!-- INNER JOIN [dbo].[ivRecording] ivR ON ivR.SurveyId = ivS.Id -->
<!-- -- Deel met soortenlijst en synoniem -->
<!-- INNER JOIN [dbo].[ivRLayer] ivRL_Layer on ivRL_Layer.RecordingID = ivR.Id -->
<!-- INNER JOIN [dbo].[ivRLTaxonOccurrence] ivRL_Taxon on ivRL_Taxon.LayerID = ivRL_Layer.ID -->
<!-- INNER JOIN [dbo].[ivRLIdentification] ivRL_Iden on ivRL_Iden.OccurrenceID = ivRL_Taxon.ID -->
<!-- LEFT JOIN (SELECT ftTaxon.TaxonName AS TaxonFullText -->
<!-- , COALESCE([GetSyn].TaxonName, ftTaxon.TaxonName) AS ScientificName -->
<!-- , COALESCE([GetSyn].TaxonGIVID, ftTaxon.TaxonGIVID) AS TAXON_LIST_ITEM_KEY -->
<!-- , COALESCE([GetSyn].TaxonQuickCode, ftTaxon.TaxonQuickCode) AS QuickCode -->
<!-- FROM [syno].[Futon_dbo_ftTaxon] ftTaxon -->
<!-- INNER JOIN [syno].[Futon_dbo_ftTaxonListItem] ftTLI ON ftTLI.TaxonGIVID =
ftTaxon.TaxonGIVID -->
<!-- LEFT JOIN (SELECT ftTaxonLI.TaxonListItemGIVID -->
<!-- , ftTaxon.TaxonGIVID -->
<!-- , ftTaxon.TaxonName -->
<!-- , ftTaxon.TaxonQuickCode -->
<!-- , ftAGL.ListName -->

```

```

<!-- , ftTaxonLI.PreferedListItemGIVID -->
<!-- FROM [syno].[Futon_dbo_ftActionGroupList] ftAGL -->
<!-- INNER JOIN [syno].[Futon_dbo_ftTaxonListItem] ftTaxonLI ON
ftTaxonLI.TaxonListGIVID = ftAGL.ListGIVID -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftTaxon] ftTaxon ON ftTaxon.TaxonGIVID =
ftTaxonLI.TaxonGIVID -->
<!-- WHERE 1=1 -->
<!-- AND ftAGL.ListName = 'INBO-2011 Sci' -->
<!-- ) GetSyn ON GetSyn.TaxonListItemGIVID = ftTLI.PreferedListItemGIVID -->
<!-- WHERE ftTLI.TaxonListGIVID = 'TL2011092815101010' -->
<!-- ) Synoniem on ivRL_Iden.TaxonFullText = Synoniem.TaxonFullText collate Latin1_General_CI_AI
-->
<!-- -- Hier begint deel met bedekking -->
<!-- LEFT JOIN [dbo].[ivRLResources] ivRL_Res on ivRL_Res.ResourceGIVID = ivRL_Taxon.CoverageResource -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftActionGroupList] ftAGL on ftAGL.ActionGroup = ivRL_Res.ActionGroup
collate Latin1_General_CI_AI -->
<!-- AND ftAGL.ListName = ivRL_Res.ListName collate Latin1_General_CI_AI -->
<!-- LEFT JOIN [syno].[Futon_dbo_ftCoverValues] ftCover on ftCover.ListGIVID = ftAGL.ListGIVID -->
<!-- AND ivRL_Taxon.CoverageCode = ftCover.Code collate Latin1_General_CI_AI -->
<!-- --WHERE ivR.NeedsWork = 0 -->
<!-- AND ivRL_Iden.Preferred = 1 -->
<!-- -- AND ivR.RecordingGivid = 'IV2014070310423184' --(dees bevat Betula pubescens Ehrh., in inboveg is
prefered Betula alba L. -->
<!-- AND ivS.Name LIKE {SurveyName}", -->
<!-- ivS.Name = Name, -->
<!-- .con = con)) -->
<!-- } -->
<!-- `` -->
<!-- # Example -->
<!-- SurveyName <- "OudeLanden_1979" -->
<!-- OudeLanden <- relevé_info_surveynome(SurveyName, con) -->
<!-- OudeLanden2 <- relevé_info_surveynome("OudeLanden_1979", con) -->
<!-- # Connection -->
<!-- To close the connection: -->
<!-- ``{r} -->
<!-- dbDisconnect(con) -->
<!-- rm(con) -->
<!-- `` -->

```