

Package: ciecl (via r-universe)

May 26, 2026

Title International Classification of Diseases 'ICD-10'/'ICD-11' for Chile

Version 0.9.8

Description Tools for working with the International Classification of Diseases ('ICD-10' Chile official 'MINSAL'/'DEIS' v2018). Includes optimized 'SQL' search with 'SQLite', fuzzy matching of medical terms ('Jaro-Winkler'), Charlson and Elixhauser comorbidity calculation, 'WHO' 'ICD-11' 'APT' integration, and hierarchical code validation. Data from Centro FIC Chile 'DEIS' <<https://deis.minsal.cl/centrofic/>>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

Depends R (>= 4.1.0)

Imports DBI, RSQLite, stringdist, stringr, dplyr, tibble, tools, utils, lifecycle, rlang, httr2

Suggests comorbidity, gt, readxl, usethis, withr, writexl, testthat (>= 3.1.5), knitr, litedown (>= 0.9), rmarkdown

VignetteBuilder knitr

URL <https://rodotasso.github.io/ciecl/>,
<https://github.com/RodoTasso/ciecl>

BugReports <https://github.com/RodoTasso/ciecl/issues>

Config/testthat/edition 3

Config/Needs/website pkgdown

Language en, es

RoxygenNote 7.3.2

Config/pak/sysreqs libicu-dev libssl-dev

Repository <https://rodotasso.r-universe.dev>

Date/Publication 2026-05-25 18:07:31 UTC

RemoteUrl <https://github.com/rodotasso/ciecl>

RemoteRef HEAD

RemoteSha d4076acfaf7e1cfb57f5a9d5c3e3ed4c597d51ad

Contents

cie_comorbid	2
cie_describe	3
cie_expand	4
cie_guide	5
cie_lookup	6
cie_map_comorbid	7
cie_norm	8
cie_search	9
cie_short	11
cie_table	12
cie_validate_vector	13
cie10_cl	13
cie10_clear_cache	14
cie10_disconnect	15
cie10_sql	16
cie11_search	16
Index	18

cie_comorbid	<i>Calcular comorbilidades Charlson/Elixhauser para Chile</i>
--------------	---

Description

Calcular comorbilidades Charlson/Elixhauser para Chile

Usage

```
cie_comorbid(data, id, code, map = c("charlson", "elixhauser"), assign0 = TRUE)
```

Arguments

data	data.frame con columnas id paciente + codigos CIE-10
id	String nombre columna identificador paciente
code	String nombre columna con codigos CIE-10 (uno por fila)
map	Character, esquema comorbilidad ("charlson" o "elixhauser")
assign0	Logical, asignar 0 si sin comorbilidad (default TRUE)

Value

data.frame ancho con scores comorbilidad por paciente

See Also

[cie_map_comorbid\(\)](#), [cie_norm\(\)](#)

Other comorbilidades: [cie_map_comorbid\(\)](#)

Examples

```
# Ver documentacion de parametros
args(cie_comorbid)

df <- data.frame(
  id_pac = c(1, 1, 2, 2),
  diag = c("E11.0", "I21.0", "C50.9", "E10.9")
)
cie_comorbid(df, id = "id_pac", code = "diag", map = "charlson")
```

cie_describe

Obtener descripcion de codigos CIE-10 (vector)

Description

Devuelve un vector character con la descripcion de cada codigo, pensado para usar dentro de `dplyr::mutate()` sin necesidad de un `left_join` contra `cie10_cl`.

Usage

```
cie_describe(
  codes,
  normalize = FALSE,
  default = NA_character_,
  codigos = lifecycle::deprecated()
)
```

Arguments

<code>codes</code>	Character vector de codigos CIE-10 (ej. "E11.0", <code>c("E11.0", "I10")</code>).
<code>normalize</code>	Logical, ¿intentar normalizar los codigos antes de buscar la descripcion? (default FALSE). Usar TRUE para limpiar formatos (ej. "E110" -> "E11.0"); usar FALSE para auditar la calidad original del registro.
<code>default</code>	Valor devuelto cuando un codigo no se encuentra en el catalogo. Default <code>NA_character_</code> .
<code>codigos</code>	[Deprecated] Use <code>codes</code> .

Value

Character vector del mismo largo que codes con la descripción oficial MINSAL/DEIS. NA_character_ (o default) para códigos sin match.

See Also

[cie_lookup\(\)](#) para resultado como tibble con todas las columnas; [cie_norm\(\)](#) para normalización.

Other búsqueda: [cie_guia_búsqueda\(\)](#), [cie_guide\(\)](#), [cie_lookup\(\)](#), [cie_search\(\)](#), [cie_short\(\)](#), [cie_siglas\(\)](#)

Examples

```
# Auditoria: buscar tal cual (E110 no existe sin punto)
cie_describe("E110", normalize = FALSE)

# Rescate: normalizar antes de buscar
cie_describe("E110", normalize = TRUE)

# Uso típico en auditoria VIU (contar fallos de origen)
diags <- c("E11.0", "E110", "I10X", "INVALIDO")
descripciones <- cie_describe(diags, normalize = FALSE)
sum(is.na(descripciones)) # Detecta 3 errores de registro
```

 cie_expand

Expandir código jerárquico (ej. E11 -> E11.0-E11.9)

Description

Expandir código jerárquico (ej. E11 -> E11.0-E11.9)

Expandir código jerárquico (ej. E11 -> E11.0-E11.9)

Usage

```
cie_expand(code, codigo = lifecycle::deprecated())
```

```
cie_expand(code, codigo = lifecycle::deprecated())
```

Arguments

code String código padre (ej. "E11")

codigo **[Deprecated]** Use code.

Value

Character vector con todos los codigos hijos del codigo padre. Vector vacio si el codigo no existe en la base de datos.

Character vector con todos los codigos hijos del codigo padre. Vector vacio si el codigo no existe en la base de datos.

See Also

[cie_norm\(\)](#), [cie_lookup\(\)](#)

[cie_norm\(\)](#), [cie_lookup\(\)](#)

Other validacion: [cie_norm\(\)](#), [cie_normalizar\(\)](#), [cie_validate_vector\(\)](#)

Other validacion: [cie_norm\(\)](#), [cie_normalizar\(\)](#), [cie_validate_vector\(\)](#)

Examples

```
cie_expand("E11")
cie_expand("E11")
```

cie_guide

Guia de funciones de busqueda CIE-10

Description

Muestra tabla comparativa de cuando usar cada funcion de busqueda.

Usage

```
cie_guide()
```

Value

tibble con guia comparativa de funciones de busqueda

See Also

[cie_search\(\)](#), [cie_lookup\(\)](#), [cie_short\(\)](#)

Other busqueda: [cie_describe\(\)](#), [cie_guia_busqueda\(\)](#), [cie_lookup\(\)](#), [cie_search\(\)](#), [cie_short\(\)](#), [cie_siglas\(\)](#)

Examples

```
cie_guide()
```

 cie_lookup

Busqueda exacta por codigo CIE-10

Description

Busqueda exacta por codigo CIE-10

Usage

```

cie_lookup(
  code,
  expand = FALSE,
  normalize = TRUE,
  full_description = FALSE,
  extract = FALSE,
  check_siglas = FALSE,
  codigo = lifecycle::deprecated(),
  expandir = lifecycle::deprecated(),
  normalizar = lifecycle::deprecated(),
  descripcion_completa = lifecycle::deprecated()
)

```

Arguments

code	Character vector de codigos (ej. "E11", "E11.0", c("E11.0", "Z00")) o rango (ej. "E10-E14"). Acepta vectores. Soporta formatos: con punto (E11.0), sin punto (E110), o solo categoria (E11).
expand	Logical, expandir jerarquia completa (default FALSE)
normalize	Logical, normalizar formato de codigos automaticamente (default TRUE)
full_description	Logical, agregar columna descripcion_completa con formato "CODIGO - DESCRIPCION" (default FALSE)
extract	Logical, extraer codigo CIE-10 de texto con prefijos/sufijos (default FALSE). IMPORTANTE: Solo usar con codigo ESCALAR (longitud 1). Ejemplo: "CIE:E11.0" -> "E11.0", "E11.0-confirmado" -> "E11.0". Para vectores multiples usar extract=FALSE (default).
check_siglas	Logical, buscar siglas medicas comunes (default FALSE). Ejemplo: "IAM" -> I21.0 (Infarto agudo miocardio)
codigo	[Deprecated] Use code.
expandir	[Deprecated] Use expand.
normalizar	[Deprecated] Use normalize.
descripcion_completa	[Deprecated] Use full_description.

Value

tibble con código(s) matcheado(s)

See Also

[cie_search\(\)](#), [cie_norm\(\)](#), [cie_expand\(\)](#)

Other búsqueda: [cie_describe\(\)](#), [cie_guia_búsqueda\(\)](#), [cie_guide\(\)](#), [cie_search\(\)](#), [cie_short\(\)](#), [cie_siglas\(\)](#)

Examples

```
# Búsqueda directa por código
cie_lookup("E11.0")

cie_lookup("E110")      # Sin punto
cie_lookup("E11")       # Solo categoría
cie_lookup("E11", expand = TRUE) # Todos E11.x
# Vectorizado - múltiples códigos y formatos
cie_lookup(c("E11.0", "Z00", "I10"))
# Con descripción completa
cie_lookup("E110", full_description = TRUE)
# Extraer código de texto con ruido (solo código escalar)
cie_lookup("CIE:E11.0", extract = TRUE)
cie_lookup("E11.0-confirmado", extract = TRUE)
# Buscar por siglas médicas
cie_lookup("IAM", check_siglas = TRUE)
cie_lookup("DM2", check_siglas = TRUE)
```

cie_map_comorbid *Mapeo manual grupos comorbilidad Chile-específico*

Description

Agrupar códigos CIE-10 chilenos en categorías comorbilidad MINSAL. Basado en Decreto 1301/2016 MINSAL + icd::icd10_map_charlson.

Usage

```
cie_map_comorbid(codes, codigos = lifecycle::deprecated())
```

Arguments

codes Character vector de códigos
codigos **[Deprecated]** Use codes.

Value

tibble con columnas: codigo, categoria

See Also

[cie_comorbid\(\)](#), [cie_norm\(\)](#)

Other comorbilidades: [cie_comorbid\(\)](#)

Examples

```
cie_map_comorbid(c("E11.0", "I50.9", "C50.9"))
```

cie_norm

Normalizar codigos CIE-10 a formato con punto

Description

Convierte codigos CIE-10 de diferentes formatos al formato estandar (con punto). Maneja multiples variaciones de entrada comunes en datos clinicos.

Usage

```
cie_norm(  
  codes,  
  search_db = TRUE,  
  codigos = lifecycle::deprecated(),  
  buscar_db = lifecycle::deprecated()  
)
```

```
cie_normalize(  
  codes,  
  search_db = TRUE,  
  codigos = lifecycle::deprecated(),  
  buscar_db = lifecycle::deprecated()  
)
```

Arguments

codes	Character vector de codigos en cualquier formato
search_db	Logical, buscar codigo en base de datos si no se encuentra exacto (default TRUE)
codigos	[Deprecated] Use codes.
buscar_db	[Deprecated] Use search_db.

Details

La normalizacion incluye:

- Conversion a mayusculas
- Eliminacion de espacios (inicio, fin e internos)
- Eliminacion de simbolos daga y asterisco (codificacion dual)
- Conversion de guiones a puntos (I10-0 -> I10.0)
- Eliminacion de puntos iniciales (.I10 -> I10)
- Correccion de puntos multiples (E..11 -> E.11)
- Eliminacion de sufijo X en codigos cortos (I10X -> I10)
- Preservacion de X en codigos largos (placeholder 7o caracter)
- Agregado de punto en posicion correcta (E110 -> E11.0)

El sistema de daga/asterisco indica codificacion dual donde la daga marca la enfermedad subyacente y el asterisco la manifestacion. Ambos simbolos se eliminan para normalizacion.

Value

Character vector con codigos normalizados al formato con punto

See Also

[cie_validate_vector\(\)](#), [cie_expand\(\)](#), [cie_lookup\(\)](#)

Other validacion: [cie_expand\(\)](#), [cie_normalizar\(\)](#), [cie_validate_vector\(\)](#)

Examples

```

cie_norm("E110")      # Retorna "E11.0"
cie_norm("E11")       # Retorna "E11" (categoria)
cie_norm("I10X")      # Retorna "I10" (elimina X)
cie_norm("E 11 0")    # Retorna "E11.0" (espacios internos)
cie_norm("I10-0")     # Retorna "I10.0" (guion a punto)
cie_norm(paste0("A17.0", intToUtf8(0x2020))) # "A17.0" (elimina daga)
cie_norm("G01*")      # "G01" (elimina asterisco)
cie_norm(c("E110", "I10X", "Z00")) # Vectorizado

```

cie_search

Busqueda difusa (fuzzy) de terminos medicos CIE-10

Description

Busca en descripciones CIE-10 usando multiples estrategias:

1. Expansion de siglas medicas (IAM, TBC, DM, etc.)
2. Busqueda exacta por subcadena (mas rapida)
3. Busqueda fuzzy con Jaro-Winkler (tolera typos)

Usage

```

cie_search(
  text,
  threshold = 0.7,
  max_results = 50,
  field = c("descripcion", "inclusion"),
  only_fuzzy = FALSE,
  verbose = TRUE,
  texto = lifecycle::deprecated(),
  campo = lifecycle::deprecated(),
  solo_fuzzy = lifecycle::deprecated()
)

```

Arguments

text	String termino medico en espanol o sigla (ej. "diabetes", "IAM", "TBC")
threshold	Numeric entre 0 y 1, umbral similitud Jaro-Winkler (default 0.70)
max_results	Integer, maximo resultados a retornar (default 50)
field	Character, campo busqueda ("descripcion" o "inclusion")
only_fuzzy	Logical, usar solo busqueda fuzzy sin busqueda exacta (default FALSE)
verbose	Logical, mostrar mensajes informativos (default TRUE). Usar FALSE en scripts.
texto	[Deprecated] Use text.
campo	[Deprecated] Use field.
solo_fuzzy	[Deprecated] Use only_fuzzy.

Details

La busqueda es tolerante a tildes: "neumonia" encuentra "neumonia". Soporta siglas medicas comunes: "IAM" busca "infarto agudo miocardio".

Value

tibble ordenado por score descendente (1.0 = coincidencia exacta). Si el text corresponde a una sigla medica, se expande automaticamente antes de buscar.

See Also

[cie_lookup\(\)](#), [cie_short\(\)](#), [cie10_sql\(\)](#)

Other busqueda: [cie_describe\(\)](#), [cie_guia_busqueda\(\)](#), [cie_guide\(\)](#), [cie_lookup\(\)](#), [cie_short\(\)](#), [cie_siglas\(\)](#)

Examples

```

# Busqueda basica
cie_search("diabetes")

```

```

cie_search("neumonia")

# Búsqueda por siglas medicas
cie_search("IAM")
cie_search("DM2")

# Tolerante a tildes y typos
cie_search("diabetis")

# Buscar en inclusiones
cie_search("bacteriana", field = "inclusion")

```

cie_short

Listar siglas medicas soportadas

Description

Muestra todas las siglas medicas que pueden usarse en [cie_search\(\)](#). "Sigla" se conserva como termino local (concepto medico chileno) en la columna de salida; el nombre de la funcion usa `cie_short` por consistencia con el ecosistema R (verbos cortos en ingles).

Usage

```
cie_short(category = NULL, categoria = lifecycle::deprecated())
```

Arguments

<code>category</code>	Character opcional, filtrar por categoria. Valores validos: "cardiovascular", "respiratoria", "metabolica", "gastrointestinal", "infecciosa", "oncologica", "reumatologica", "neurologica", "psiquiatrica", "traumatologica", "pediatrica", "gineco_obstetrica". Si es NULL (default), retorna todas las siglas.
<code>categoria</code>	[Deprecated] Use category.

Value

tibble con columnas: sigla, termino_búsqueda, categoria

See Also

[cie_search\(\)](#), [cie_lookup\(\)](#)

Other búsqueda: [cie_describe\(\)](#), [cie_guia_búsqueda\(\)](#), [cie_guide\(\)](#), [cie_lookup\(\)](#), [cie_search\(\)](#), [cie_siglas\(\)](#)

Examples

```
# Ver todas las siglas
cie_short()

# Filtrar por categoria
cie_short("cardiovascular")
cie_short("oncologica")

# Buscar una sigla especifica
cie_short() |> dplyr::filter(sigla == "iam")
```

cie_table

Generar tabla HTML interactiva GT de codigo CIE-10

Description

Muestra la jerarquia de un codigo CIE-10 (categoria + subcategorias) como una tabla gt. Las columnas "Incluye" y "Excluye" pueden aparecer vacias en subcategorias: el catalogo MINSAL/DEIS no puebla esos campos en todos los niveles (suelen estar solo en la categoria de 3 digitos). Para evitar confusion visual, los NA se reemplazan por un guion largo (em dash).

Usage

```
cie_table(code, codigo = lifecycle::deprecated())
```

Arguments

code String codigo (ej. "E11" muestra la jerarquia).
codigo **[Deprecated]** Use code.

Value

Objeto de clase gt_tbl (tabla HTML interactiva).

See Also

[cie_search\(\)](#), [cie_lookup\(\)](#)

Examples

```
cie_table("E11") # Diabetes mellitus tipo 2 completo
```

cie_validate_vector *Validar vector de codigos CIE-10 formato*

Description

Validar vector de codigos CIE-10 formato

Usage

```
cie_validate_vector(codes, strict = FALSE, codigos = lifecycle::deprecated())
```

Arguments

codes	Character vector codigos (ej. c("E11.0", "Z00.0"))
strict	Logical, validar existencia en DB (default FALSE)
codigos	[Deprecated] Use codes.

Value

Logical vector de la misma longitud que codes. TRUE si el codigo tiene formato CIE-10 valido (y existe en DB si strict = TRUE).

See Also

[cie_norm\(\)](#), [cie_expand\(\)](#)

Other validacion: [cie_expand\(\)](#), [cie_norm\(\)](#), [cie_normalizar\(\)](#)

Examples

```
cie_validate_vector(c("E11.0", "INVALIDO", "Z00"))
```

cie10_cl *Dataset CIE-10 Chile oficial MINSAL/DEIS v2018*

Description

Dataset CIE-10 Chile oficial MINSAL/DEIS v2018

Usage

```
cie10_cl
```

Format

tibble con 39,877 filas (categorias y subcategorias):

codigo Codigo CIE-10 (ej. "E11.0")
descripcion Diagnostico en espanol chileno
categoria Categoria jerarquica
seccion Seccion dentro del capitulo
capitulo_nombre Nombre descriptivo del capitulo
inclusion Terminos incluidos
exclusion Terminos excluidos
capitulo Capitulo CIE-10 (A-Z)
es_daga Logical, codigo daga (+)
es_cruz Logical, codigo asterisco (*)

Source

<https://deis.minsal.cl/centrofic/>

Examples

```
data(cie10_cl)
head(cie10_cl)
```

cie10_clear_cache	<i>Limpiar cache SQLite (forzar rebuild)</i>
-------------------	--

Description

Limpiar cache SQLite (forzar rebuild)

Usage

```
cie10_clear_cache()
```

Value

No return value, called for side effects (deletes SQLite cache).

See Also

[cie10_sql\(\)](#), [cie10_disconnect\(\)](#)
Other sql: [cie10_disconnect\(\)](#), [cie10_sql\(\)](#)

Examples

```
# Ver ubicacion del cache
tools::R_user_dir("ciecl", "data")

cie10_clear_cache() # Elimina cie10.db local
```

cie10_disconnect	<i>Cerrar conexion pooled SQLite</i>
------------------	--------------------------------------

Description

Cierra la conexion reutilizable al archivo SQLite. Util para liberar el lock del archivo .db.

Usage

```
cie10_disconnect()
```

Value

No return value, called for side effects.

See Also

[cie10_sql\(\)](#), [cie10_clear_cache\(\)](#)

Other sql: [cie10_clear_cache\(\)](#), [cie10_sql\(\)](#)

Examples

```
# Verificar si hay conexion activa
is.null(ciecl:::ciecl_env$con)

cie10_disconnect()
```

`cie10_sql`*Ejecutar consultas SQL sobre CIE-10 Chile*

Description

Ejecutar consultas SQL sobre CIE-10 Chile

Usage

```
cie10_sql(query, close = lifecycle::deprecated())
```

Arguments

<code>query</code>	String SQL valido SQLite (SELECT/WHERE/JOIN)
<code>close</code>	[Deprecated] Ignorado — la conexion es pooled y se gestiona automaticamente. Sera eliminado en una version futura.

Value

tibble resultado query

See Also

[cie10_clear_cache\(\)](#), [cie10_disconnect\(\)](#), [cie_search\(\)](#)

Other sql: [cie10_clear_cache\(\)](#), [cie10_disconnect\(\)](#)

Examples

```
# Buscar diabetes
cie10_sql("SELECT codigo, descripcion FROM cie10 WHERE codigo LIKE 'E11%'")
```

```
# Contar por capitulo
cie10_sql("SELECT capitulo, COUNT(*) n FROM cie10 GROUP BY capitulo")
```

`cie11_search`*Buscar codigos CIE-11 via API OMS*

Description

Buscar codigos CIE-11 via API OMS

Usage

```
cie11_search(  
  text,  
  api_key = NULL,  
  lang = "es",  
  max_results = 10,  
  release = "2024-01",  
  texto = lifecycle::deprecated()  
)
```

Arguments

text	String termino busqueda espanol/ingles
api_key	String opcional, Client ID + Secret OMS separados ":" Obtener en: https://icd.who.int/icdapi
lang	Character, idioma respuesta ("es" o "en")
max_results	Integer, maximo resultados (default 10)
release	Character, version de release CIE-11 a consultar (default "2024-01"). Ver releases disponibles en la API OMS.
texto	[Deprecated] Use text.

Value

tibble con codigos CIE-11 + titulos o vacio si error

See Also

[cie_search\(\)](#), [cie_lookup\(\)](#)

Examples

```
# Ver parametros disponibles  
args(cie11_search)  
  
# Requiere credenciales OMS gratuitas (https://icd.who.int/icdapi)  
Sys.setenv(ICD_API_KEY = "client_id:client_secret")  
cie11_search("depression mayor")
```

Index

- * **api**
 - cie11_search, 16
- * **busqueda**
 - cie_describe, 3
 - cie_guide, 5
 - cie_lookup, 6
 - cie_search, 9
 - cie_short, 11
- * **comorbilidades**
 - cie_comorbid, 2
 - cie_map_comorbid, 7
- * **datasets**
 - cie10_cl, 13
- * **sql**
 - cie10_clear_cache, 14
 - cie10_disconnect, 15
 - cie10_sql, 16
- * **validacion**
 - cie_expand, 4
 - cie_norm, 8
 - cie_validate_vector, 13
- * **visualizacion**
 - cie_table, 12

- cie10_cl, 13
- cie10_clear_cache, 14, 15, 16
- cie10_clear_cache(), 15, 16
- cie10_disconnect, 14, 15, 16
- cie10_disconnect(), 14, 16
- cie10_sql, 14, 15, 16
- cie10_sql(), 10, 14, 15
- cie11_search, 16
- cie_comorbid, 2, 8
- cie_comorbid(), 8
- cie_describe, 3, 5, 7, 10, 11
- cie_expand, 4, 9, 13
- cie_expand(), 7, 9, 13
- cie_guia_busqueda, 4, 5, 7, 10, 11
- cie_guide, 4, 5, 7, 10, 11
- cie_lookup, 4, 5, 6, 10, 11
- cie_lookup(), 4, 5, 9-12, 17
- cie_map_comorbid, 3, 7
- cie_map_comorbid(), 3
- cie_norm, 5, 8, 13
- cie_norm(), 3-5, 7, 8, 13
- cie_normalizar, 5, 9, 13
- cie_normalize (cie_norm), 8
- cie_search, 4, 5, 7, 9, 11
- cie_search(), 5, 7, 11, 12, 16, 17
- cie_short, 4, 5, 7, 10, 11
- cie_short(), 5, 10
- cie_siglas, 4, 5, 7, 10, 11
- cie_table, 12
- cie_validate_vector, 5, 9, 13
- cie_validate_vector(), 9