



Implementación del estándar de evaluación de calidad en encuestas económicas: fundamentos y aplicación práctica en R mediante el paquete calidad

Área de Ciencia de Datos

Unidad de Gobierno de Datos

Agosto 2025



Objetivos de la presentación

- Dar a conocer las nuevas funcionalidades del paquete calidad
- Demostrar cómo implementar el estándar para encuestas económicas

Antes de comenzar

⚠ La versión actual del paquete siempre puede mejorar ⚠

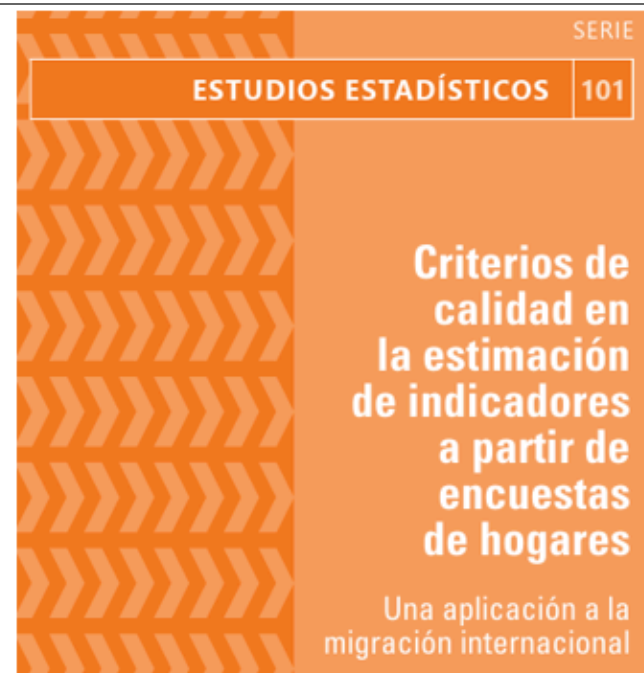
Quedamos atentos a su feedback 😊

Paquete calidad



Estándar para la evaluación de la calidad de las estimaciones en encuestas de hogares

Recomendaciones metodológicas
sobre la medición de la calidad
de las cifras provenientes
de encuestas de hogares



Estándar para la evaluación de la calidad de las estimaciones en encuestas económicas





Estándar de hogares INE

Estándar de encuestas económicas INE

Estándar CEPAL

Estándar regional

Paquete calidad | Flujo

- Dos grandes funcionalidades:

1. Realizar la estimación

- Funciones *create_prop()*, *create_mean()*, *create_size()* y *create_total()*

2. Realizar la evaluación de la estimación (*función assess()*)

- *calidad* se apoya en el paquete *survey* para realizar las estimaciones



Paquete calidad | Funciones *create_**

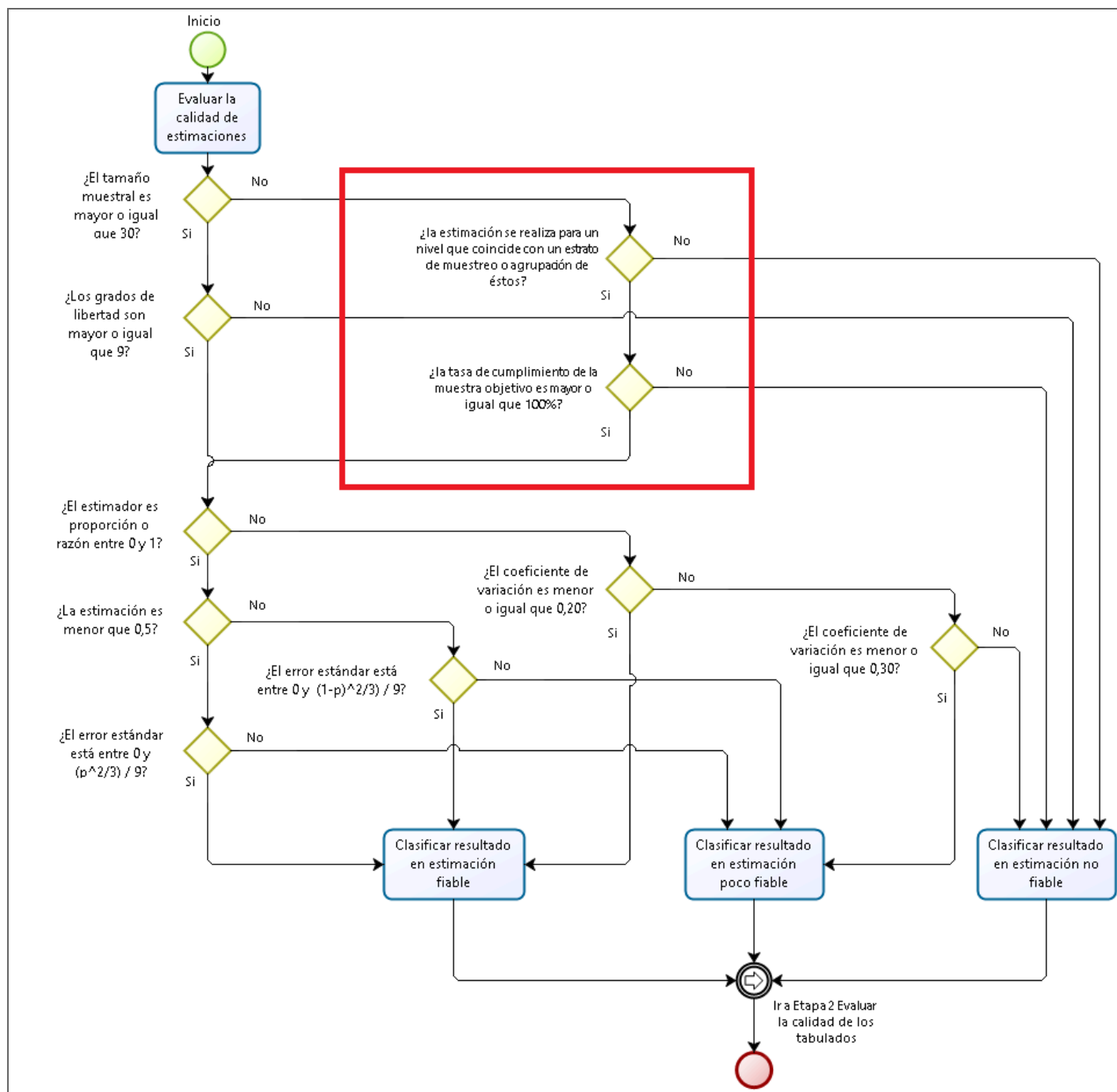
- Las funciones *create_()* generan los insumos necesarios para realizar la evaluación.
- Los parámetros principales son:
 - *var* : Variable a estimar
 - *domains*: Dominio de interés.
 - *subpop*: Subpoblación de referencia (variable *dummy* que realiza filtro de los datos).
 - *design*: Diseño de muestra creado con *survey*
- Por defecto nos retorna los insumos: **se, df, n y cv**.
- Adicionalmente, estas funciones nos pueden retornar el **deff, ess, log_cv y ci**.

Paquete calidad | Función *assess*

- Función *assess()*: evaluación del estándar
- **Parámetros principales:**
 1. *table*: Tabla con las estimaciones generadas con funciones *create_...()*.
 2. *scheme*: String que indica el estándar a evaluar, este puede ser “chile”, “eclac_2020”, “eclac_2023” y “**chile_economics**”.
 3. *publish*: Booleano, *TRUE* evalúa el porcentaje de estimaciones fiables en el tabulado completo. Default *FALSE*.
 4. *ratio_between_0_1*: Booleano, *TRUE* indica que la estimación corresponde a una proporción o razón entre 0 y 1. Por defecto es **TRUE** y se evalúa solo cuando tenemos una clase proporción (estimación con *create_prop*).

Estándar de calidad para encuestas económicas (1/3)

Recordemos un poco el estándar de encuestas económicas:



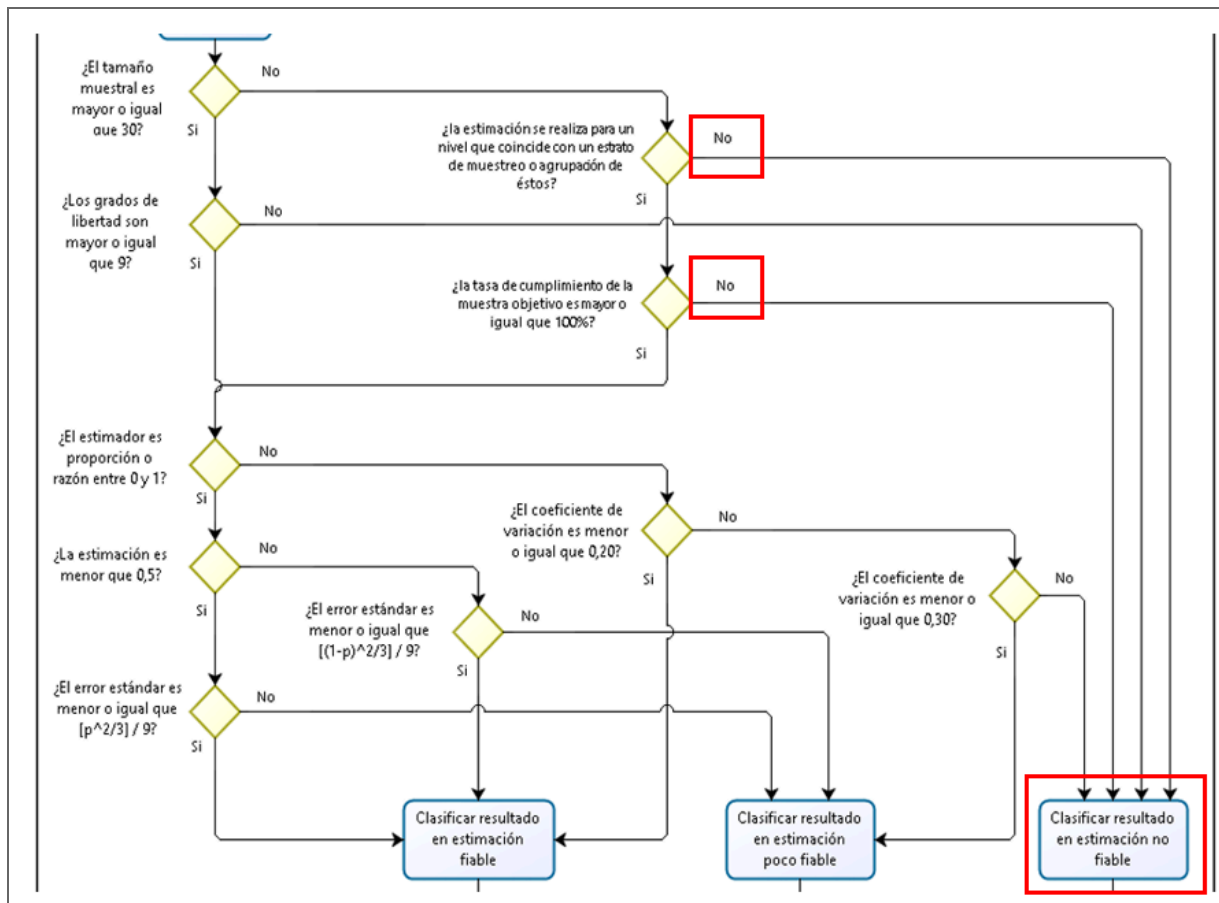


Estándar de calidad para encuestas económicas (2/3)

- Parámetros particulares para el estándar de económicas en *assess()*:
 1. **domain_info**: *TRUE* implica que la estimación coincide con un estrato de muestreo. Default *FALSE*.
 2. **table_n_obj**: Dataframe que contiene el tamaño de muestra objetivo (**n_obj**) de los dominios indicados en la estimación.
 - ⚠ Es importante que en este dataframe los nombres de los dominios **coincidan** con los utilizados en la función *create_()* y que la columna del tamaño objetivo se llame **n_obj**.

Estándar de calidad para encuestas económicas (3/3)

⚠ Si no hay información, el paquete tomará una decisión conservadora ⚠



Aplicación

- Usaremos los datos de la **VII ELE**, disponibles en el paquete
- Estimaremos y evaluaremos la productividad salarial (razón entre el Valor agregado 2022 y Remuneración total bruta del personal contratado) por tamaño y actividad.
- ⚠ No soy experto en la ELE ⚠

¡Vamos a R 🚀!

Primeros pasos

Debemos instalar calidad y cargar algunas dependencias

```
1 install.packages('calidad')
```

```
1 library(calidad)
2 library(survey)   # para crear diseño de muestra
3 library(dplyr)
```


Cargando datos

El paquete cuenta con datos de prueba de la ELE7 (Encuesta Longitudinal de Empresas)

```
1 head(ELE7, 3) %>%
2   kableExtra::kbl() %>%
3   kableExtra::kable_styling(font_size = 15)
```

rol_ficticio	cod_actividad	cod_tamano	tramo	fe_transversal	fe_longitudinal	panel	estrato	pob	
303008	G	1	2	9.205626	0.00000	0	G12	4963.552	92
109439	F	1	2	12.673940	17.98274	1	F12	1163.723	1268
401684	L-N	3	2	116.554955	105.33128	1	L- N32	8322.855	19

Declaramos el diseño

```
1 options(survey.lonely.psu = 'remove')    # indicamos tratamiento upm
2
3 ## diseño de muestra con factor de expansión transversal
4 dc_ele <- svydesign(ids = ~rol_ficticio,
5                   weights = ~fe_transversal,
6                   strata = ~estrato,
7                   fpc = ~pob,                # correccion por poblaci
8                   data = ELE7)
```

Estimando proporciones

Productividad salarial (Valor agregado / remuneración bruta de personal contratado)

```
1 ## sin dominios
2 create_prop(var= 'VA_2022f',
3             denominator = 'REMP_TOTAL',
4             design = dc_ele)
```

	stat	se	df	n	cv
1	3.888008	0.1257553	6517	6592	0.03234442

```
1 # según código de tamaño y código de actividad
2 prod_salarial <- create_prop('VA_2022f',
3                              denominator = 'REMP_TOTAL',
4                              domains = 'cod_tamano+ cod_actividad',
5                              design = dc_ele)
```

	cod_tamano	cod_actividad	stat	se	df	n	cv
1	A		2.695245	0.3691871	42	44	0.1369772

cod_tamano	cod_actividad	stat	index		df	n	cv
			se				
2	A	2.224927	0.2724771		56	57	0.1224656
3	A	2.535230	0.5201378		75	76	0.2051640
4	A	3.500658	0.9880417		74	75	0.2822446
5	A	2.981067	0.9753872		148	149	0.3271940
1	B	7.725046	0.6668554		33	35	0.0863238
2	B	6.443608	0.4141585		24	26	0.0642743
3	B	2.098264	0.6720672		51	53	0.3202967
4	B	3.371206	0.5999507		61	62	0.1779632
5	B	4.239709	0.9463727		84	85	0.2232164
1	C	3.901510	0.2800475		140	142	0.0717793
2	C	2.076863	0.2309598		51	52	0.1112061
3	C	2.219461	0.1965683		57	58	0.0885658
4	C	3.051892	0.4337962		87	88	0.1421401
5	C	1.990821	0.3424373		66	67	0.1720081
1	D-E	4.797310	1.0086230		33	35	0.2102476

cod_tamano	cod_actividad	stat	index		df	n	cv
			se				
2	D-E	4.236699	1.5120880		38	39	0.3569024
1	F	2.251525	0.1707510		101	103	0.0758380
2	F	2.020970	0.0986236		157	158	0.0488002
3	F	2.074069	0.2632363		66	67	0.1269178
4	F	1.638795	0.2747675		54	55	0.1676643
5	F	2.738110	0.2652788		228	229	0.0968839
1	G	4.151010	0.2674126		559	561	0.0644211
2	G	3.139556	0.2221740		166	167	0.0707661
3	G	2.547756	0.2455426		142	143	0.0963760
4	G	5.706449	1.3270805		83	84	0.2325580
5	G	4.508110	1.0708710		182	183	0.2375432
1	H	4.690550	0.9198437		85	87	0.1961057
2	H	2.116024	0.1623628		29	30	0.0767301
3	H	2.523775	0.2196714		93	94	0.0870408
4	H	3.117359	0.4749287		56	57	0.1523497

cod_tamano	cod_actividad	stat	index		df	n	cv
			se				
5	H	5.208884	0.7084139	117	118	0.1360011	
1	I	2.423392	0.2126097	17	19	0.0877323	
2	I	2.415887	0.1878179	35	36	0.0777428	
3	I	2.569028	0.3333918	29	30	0.1297735	
4	I	1.761445	0.2061126	52	54	0.1170133	
5	I	3.055040	1.0842063	30	31	0.3548911	
1	J	2.621052	0.2885672	66	68	0.1100959	
2	J	1.819437	0.2514364	32	33	0.1381947	
3	J	1.474328	0.3306286	41	42	0.2242571	
4	J	2.403179	0.2536792	48	49	0.1055598	
5	J	4.294340	2.1835756	42	43	0.5084776	
1	K	10.869981	1.0269918	243	245	0.0944796	
2	K	8.366024	2.6910342	118	119	0.3216622	
1	L-N	3.848444	0.4453128	251	253	0.1157124	
2	L-N	3.295210	0.4607873	174	175	0.1398355	

cod_tamano	cod_actividad	stat	index		df	n	cv
			se				
3	L-N	2.811812	0.5858151		73	74	0.2083407
4	L-N	3.128685	0.3336174		554	555	0.1066318
5	L-N	5.442285	1.5823060		245	246	0.2907429
1	M	2.776568	0.9372290		171	173	0.3375495
2	M	1.814416	0.1163816		87	88	0.0641427
3	M	2.164218	0.1331088		120	121	0.0615043
4	M	4.060210	0.7110000		65	66	0.1751141
5	M	2.826529	0.2138901		470	471	0.0756723
1	R-S	2.416124	0.2800690		29	31	0.1159167
2	R-S	1.171757	0.2981647		37	38	0.2544596
3	R-S	2.156306	0.1946277		168	169	0.0902598
4	R-S	2.660486	0.2964359		27	28	0.1114217
5	R-S	2.433597	0.8740041		55	56	0.3591408

Nuevos pasos del estándar

Necesitamos el tamaño de muestra objetivo

El paquete cuenta con una tabla para la ELE7 😊

```
1 ELE7_n_obj %>%  
2   select(-cod_actividad) %>%  
3   head(4) %>%  
4   kableExtra::kbl() %>%  
5   kableExtra::kable_styling(font_size = 14)
```

cod_tamano	cod_actividad_letra	n_obj
1	K	358
2	K	210
1	A	54
1	L-N	234

Agregando muestra objetivo (error)

```

1 assess(prod_salarial,
2       scheme = 'chile_economics', # indicamos criterio
3       domain_info = T,           # ¿pertenece a dominio planificado?
4       table_n_obj = ELE7_n_obj)

```

	col	table1	table2	coincide
cod_tamano	cod_tamano	character	numeric	FALSE
cod_actividad	cod_actividad	character	numeric	FALSE

Error in check_type_cols(table, table_n_obj, columns): Columns types doesn't match. Please review your data.

✗ Las clases de las columnas no coinciden

```

1 str(ELE7_n_obj)

```

```

tibble [59 x 4] (S3: tbl_df/tbl/data.frame)
 $ cod_tamano      : num [1:59] 1 2 1 1 2 3 4 5 2 1 ...
 $ cod_actividad_letra: chr [1:59] "K" "K" "A" "L-N" ...

```

```

$ cod_actividad      : num [1:59] 10 10 1 11 11 11 11 11 1 12 ...
$ n_obj              : num [1:59] 358 210 54 234 216 60 736 321 54 194
...
- attr(*, "na.action")= 'omit' Named int [1:19] 1 2 3 4 5 6 7 8 9 10 ...
..- attr(*, "names")= chr [1:19] "1" "2" "3" "4" ...

```

```
1 str(prod_salarial)
```

Classes 'calidad.prop', 'svyby' and 'data.frame': 59 obs. of 7 variables:

```

$ cod_tamano      : chr  "1" "2" "3" "4" ...
$ cod_actividad: chr  "A" "A" "A" "A" ...
$ stat           : num   2.7 2.22 2.54 3.5 2.98 ...
$ se             : num   0.369 0.272 0.52 0.988 0.975 ...
$ df             : num   42 56 75 74 148 33 24 51 61 84 ...
$ n             : int   44 57 76 75 149 35 26 53 62 85 ...
$ cv             : Named num  0.137 0.122 0.205 0.282 0.327 ...
..- attr(*, "names")= chr [1:59] "1.A" "2.A" "3.A" "4.A" ...

```

Agregando muestra objetivo

Podemos modificar la tabla auxiliar para ajustarla a la salida de create

```
1 # modifiquemos la tabla ELE7_n_obj
2 ELE7_n_obj2 <- ELE7_n_obj %>%
3   mutate(cod_actividad = cod_actividad_letra,
4          cod_tamano = as.character(cod_tamano))
5
6
7 evaluacion_estandar <- assess(prod_salarial,
8                               scheme = 'chile_economics',
9                               domain_info = T,
10                              table_n_obj = ELE7_n_obj2, # tabla con cla
11                              ratio_between_0_1 = FALSE) # no es una est
```

Revisemos el resultado

```
1 # revisión del estado final para n<30:
2 evaluacion_estandar %>%
3   select(cod_tamano, cod_actividad, stat, n, compliance_rate, label) %>%
4   filter(n<30) %>%
5   kableExtra::kbl() %>%
6   kableExtra::kable_styling(font_size = 16)
```

cod_tamano	cod_actividad	stat	n	compliance_rate	label
2	B	6.443608	26	0.8125000	non-reliable
1	I	2.423392	19	0.7037037	non-reliable
4	R-S	2.660486	28	0.9032258	non-reliable

Hagamos un experimento

Modifiquemos un tamaño objetivo, para subir artificialmente la tasa de logro

```
1 ELE7_n_obj2 %>%
2   filter(cod_tamano == '2' & cod_actividad == 'B') %>%
3   kableExtra::kbl() %>%
4   kableExtra::kable_styling(font_size = 16)
```

cod_tamano	cod_actividad_letra	cod_actividad	n_obj
2	B	B	32

```
1 ELE7_n_obj3 <- ELE7_n_obj2 %>%
2   mutate(n_obj = ifelse(cod_tamano == '2' & cod_actividad == 'B', 20, n_o
```

Hagamos un experimento

```
1 evaluacion_modificada <- assess(prod_salarial,  
2                               scheme = 'chile_economics',  
3                               domain_info = T,  
4                               table_n_obj = ELE7_n_obj3,  
5                               ratio_between_0_1 = FALSE)  
6  
7 evaluacion_modificada %>%  
8   select(cod_tamano, cod_actividad, stat, n, compliance_rate, label) %>%  
9   filter(n<30) %>%  
10  kableExtra::kbl() %>%  
11  kableExtra::kable_styling(font_size = 16)
```

cod_tamano	cod_actividad	stat	n	compliance_rate	label
2	B	6.443608	26	1.3000000	reliable
1	I	2.423392	19	0.7037037	non-reliable

cod_tamano	cod_actividad	stat	n	index	
				compliance_rate	label
4	R-S	2.660486	28	0.9032258	non-reliable

Siempre puede haber problemas

¿Qué pasa si se hace imposible incluir el dataframe en la función `assess`? 😞

Podemos agregar directamente la columna `n_obj` en la salida de `create`

```
1 # tabla prod_salarial le agregamos los tamaños objetivos:
2 prod_salarial2 <- prod_salarial %>%
3   left_join(ELE7_n_obj2 %>%
4     select(-cod_actividad_letra),
5     by = c('cod_tamano', 'cod_actividad'))
6
7 # verificamos que no se hayan generado NAs:
8 is.na(prod_salarial2$n_obj) %>% sum()
```

[1] 0

```
1 # realizamos la evaluación:
2 evaluacion_estandar2 <- assess(prod_salarial2,
3   scheme = 'chile_economics',
```


4

5

```
domain_info = T,  
ratio_between_0_1 = FALSE)
```

Ausencia de muestra objetivo

¿Qué pasa si no cuento con datos de muestra objetivo? 😞

El paquete etiquetará como no fiable todas las estimaciones con $n < 30$

```
1 # No incluimos información sobre muestra objetivo
2 evaluacion_sin_nobj <- assess(prod_salarial,
3                               scheme = 'chile_economics',
4                               domain_info = T,
5                               ratio_between_0_1 = FALSE)
6
7 ## revisión del estado final para n<30
8 evaluacion_sin_nobj %>%
9   select(cod_tamano, cod_actividad, stat, n, compliance_rate, label) %>%
10  filter(n<30) %>%
11  kableExtra::kbl() %>%
12  kableExtra::kable_styling(font_size = 16)
```

cod_tamano	cod_actividad	stat	n	compliance_rate	label
2	B	6.443608	26	NA	non-reliable
1	I	2.423392	19	NA	non-reliable

cod_tamano	cod_actividad	stat	index		label
			n	compliance_rate	
4	R-S	2.660486	28	NA	non-reliable

Evaluación del tabulado

```

1  evaluacion_estandar_publish <- assess(prod_salarial,
2                                     scheme = 'chile_economics',
3                                     domain_info = TRUE,
4                                     table_n_obj = ELE7_n_obj2,
5                                     ratio_between_0_1 = FALSE,
6                                     publish = TRUE) # parámetro de pub
7
8  evaluacion_estandar_publish %>%
9    select(cod_tamano, cod_actividad, stat, n, compliance_rate, label, pub
10 head() %>%
11 kableExtra::kbl() %>%
12 kableExtra::kable_styling(font_size = 16)

```

	cod_tamano	cod_actividad	stat	n	compliance_rate	label	publication
1	A		2.695245	44	0.8148148	reliable	publish
2	A		2.224927	57	1.0555556	reliable	publish
3	A		2.535230	76	1.0555556	weakly reliable	publis

cod_tamano	cod_actividad	stat	n	compliance_rate	label	publication
4	A	3.500658	75	0.9036145	weakly reliable	publish
5	A	2.981067	149	0.8816568	non-reliable	publish
1	B	7.725046	35	0.7291667	reliable	publish

Evaluación del tabulado

```
1 evaluacion_estandar_publish %>%  
2   select(label) %>%  
3   table() %>%  
4   prop.table()
```

.

non-reliable
0.1864407

reliable weakly reliable
0.6440678 0.1694915

Para finalizar

- Para mayor información sobre el uso del paquete, visita los tutoriales en el sitio del INE [link](#) y en cran [link](#)
- Si alguien está interesado/a en colaborar en la creación de nuevas funcionalidades, puede realizarlo a través de la página github.com/inesscc/calidad  
- ¡Muchas gracias al departamento de Estadísticas Económicas y al Subdepartamento de Marcos y Muestras por la ayuda y feedback  !

Quedo atento a sus comentarios 



Implementación del estándar de evaluación de calidad en encuestas económicas: fundamentos y aplicación práctica en R mediante el paquete calidad

Área de Ciencia de Datos

Unidad de Gobierno de Datos

Agosto 2025









<https://quarto.org>

Speaker notes