

Scale Likert: inversione delle polarità

L'inversione di polarità nelle scale Likert consiste nel proporre all'interno di una scala un item con valenza “negativa” anziché “positiva”, come ad esempio

item 1 - positivo: “Leggere è uno dei miei hobby preferiti”

item 2 - negativo: “Leggere è una perdita di tempo”

L'accordo con il primo item indicherà un atteggiamento favorevole alla lettura (è d'accordo chi ama leggere); l'accordo con il secondo indicherà invece un atteggiamento negativo (è d'accordo chi non ama leggere).

Questa tecnica serve per impedire il *response set* e/o inserire domande di controllo.

 [Slides con l'analisi dell'esempio](#)

Vedi anche:

- [Scale Likert](#)
- [Scale Likert: inversione delle polarità](#)
- [Grafici a barre multiple](#)
- [Alpha di Cronbach](#)
- [Trasformare il tipo di variabile](#)

 [Funzione rev_level.R](#)

Trattamento degli item con polarità invertita

In fase di analisi dei dati però, la scala (che immaginiamo unidimensionale) dovrà essere correttamente orientata dal negativo (non ama leggere) al positivo (ama leggere). Consideriamo l'esempio alla pagina [dedicata alle scale Likert](#):

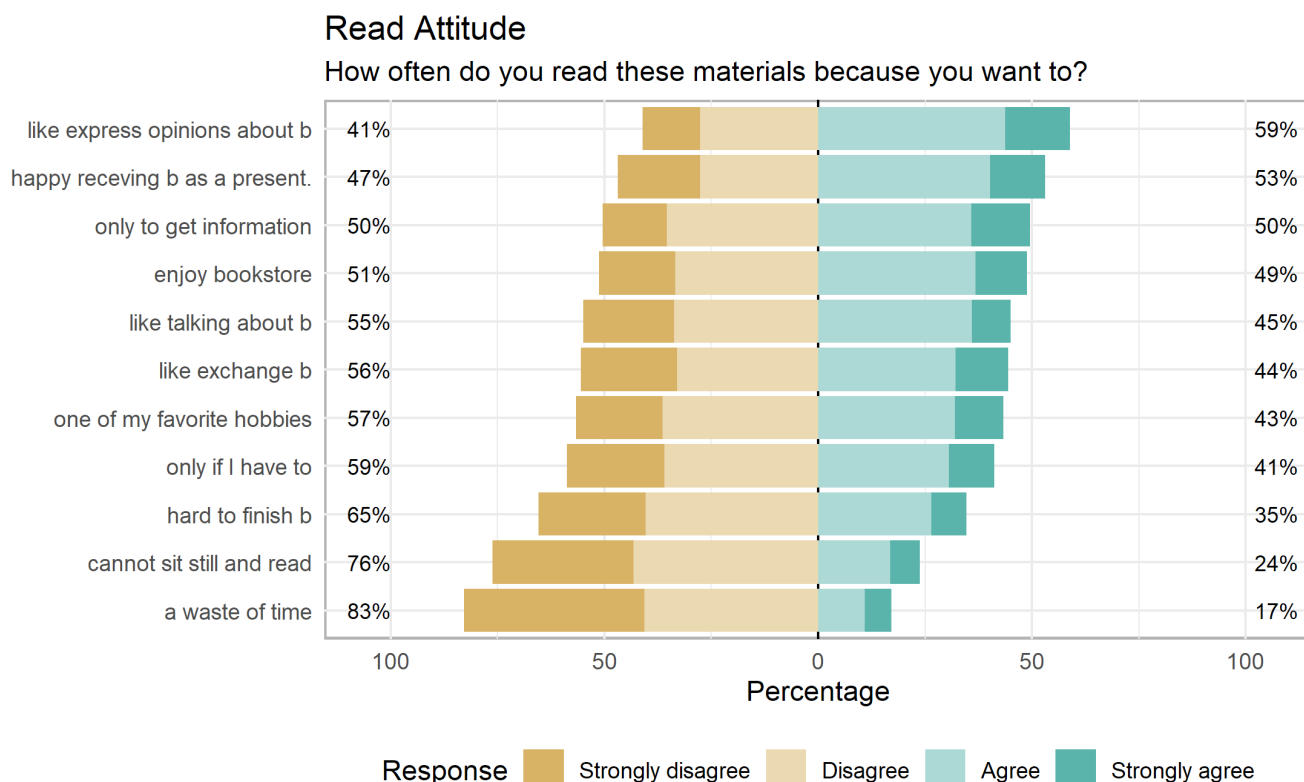


Fig. 1: Distribuzione degli items

E' evidente che i punteggi degli items "only if I have to", "hard to finish b(ooks)", "a waste of time", "cannot sit still and read" devono essere invertiti, per poter avere risultati coerenti per tutta la scala.

La presenza di item a polarità invertita è evidente dai risultati dell'**alpha (\$\alpha\$) di Cronbach**: nel caso dell'esempio, senza inversione dei punteggi, il valore di α è pari a 0,215 (molto basso), e diventa 0,890 dopo. Questo significa che prima dell'inversione dei punteggi, la scala non era coerente e unidimensionale (ovviamente).

Due sono gli approcci all'inversione dell'ordine delle modalità in R:

- **invertire solo i punteggi** (i valori numerici corrispondenti ai **livelli**) ;
- **invertire sia i livelli che le etichette** a loro associati ("Strongly disagree", "Disagree").

Per migliorare l'interpretabilità dei risultati è comunque bene **rinominare_gli_items** modificati, ad esempio assegnando un prefisso al nome.

Pacchetti e dati

[es_Likert_dati.R](#)

```
# pacchetti
```

```
library(tidyverse)
library(likert)

# dati
data(pisaitems)

# selezioniamo una sola scala come esempio
items24 <- pisaitems %>%
  select_at(vars(starts_with("ST24Q")))

# nomi degli items, per leggibilità
names(items24) <- c(ST24Q01="only if I have to",
                    ST24Q02="one of my favorite hobbies",
                    ST24Q03="like talking about b",
                    ST24Q04="hard to finish b",
                    ST24Q05="happy receving b as a present.",
                    ST24Q06="a waste of time",
                    ST24Q07="enjoy bookstore",
                    ST24Q08="to get information",
                    ST24Q09="cannot sit still and read",
                    ST24Q10="like express opinions about b",
                    ST24Q11="like exchange b")
```

Ricordiamo che i **fattori** sono memorizzati come numeri interi,

```
> typeof(items24$`only if I have to`)
```

```
[1] "integer"
```

a cui sono associati livelli, che hanno etichette di livello (*labels*):

```
> attributes(items24$`only if I have to`)
```

```
$levels
[1] "Strongly disagree" "Disagree"          "Agree"
"Strongly agree"
```

```
$class
[1] "factor"
```

Invertire solo i livelli (punteggi)

Si invertono solo i valori numerici del fattore, mantenendo così come sono le etichette dei

livelli, con la funzione `fct_rev()`, del pacchetto *forcats* (di [tidyverse](#)). Esempio:

```
library(tidyverse)

# item da modificare
table(items24$`only if I have to`)
```

Strongly disagree	Disagree	Agree	Strongly agree
14947	23515	20000	7029

I 7.029 studenti che sono fortemente d'accordo NON amano leggere.

```
# item modificato
fct_rev(items24$`only if I have to`) %>% table()
```

Strongly agree	Agree	Disagree	Strongly disagree
7029	20000	23515	14947

Invertendo solo i valori numerici, i 7.029 studenti vengono classificati come amanti della lettura, il che probabilmente non è quello che vogliamo.

Questo accade perché `fct_rev()` **inverte a tutti gli effetti l'ordine dei livelli**. Il valore numerico 1 diventa 4, il 2 diventa 3, ecc., e l'etichetta “Strongly agree” (ex punteggio 4) viene riassegnata a 1.

In questo contesto, ciò è un problema, in quanto la codifica dell'item invertito non è più coerente con quella degli altri items, e con la polarità della scala nel suo insieme.

Se gli items verranno trattati sempre ed esclusivamente come quantitativi, questo non fa problema, perché i valori numerici sono ora “corretti”. Ma se usiamo gli items (anche) come fattori ordinati, probabilmente vorremo che i 7.029 studenti che non amano leggere continuino ad essere classificati correttamente.

- **Pro:** le etichette corrispondono alle risposte che ha dato l'intervistato, e sono coerenti con la domanda posta, e il punteggio riflette correttamente l'orientamento della scala;
- **Contro:** le funzioni dedicate ai fattori (qualitative) considereranno sia l'ordinamento che i livelli. Nel caso di batterie di domande, come sono generalmente le scale, verrà meno l'associazione univoca, **per tutti gli items della scala**, fra il valore numerico del livello e l'etichetta (3 equivarrà ad “Agree” per alcuni items, e a “Disagree” per altri - quelli invertiti). E come se gli items fossero codificati diversamente.

```
# codifica degli altri items
table(items24$`one of my favorite hobbies`)
```

Strongly disagree	Disagree	Agree	Strongly agree
7487	13323	20935	23811

Il risultato potrebbe portare a risultati incongrui, come nel grafico seguente, in cui la scala sull'asse delle ascisse risulta invertito (agree → disagree, invece che disagree → agree). Il problema principale, è che l'orientamento dei valori sulle ascisse è *random*, o *ad hoc*, perché dipende da quanti e quali sono gli item invertiti, e dalla distribuzione delle risposte.

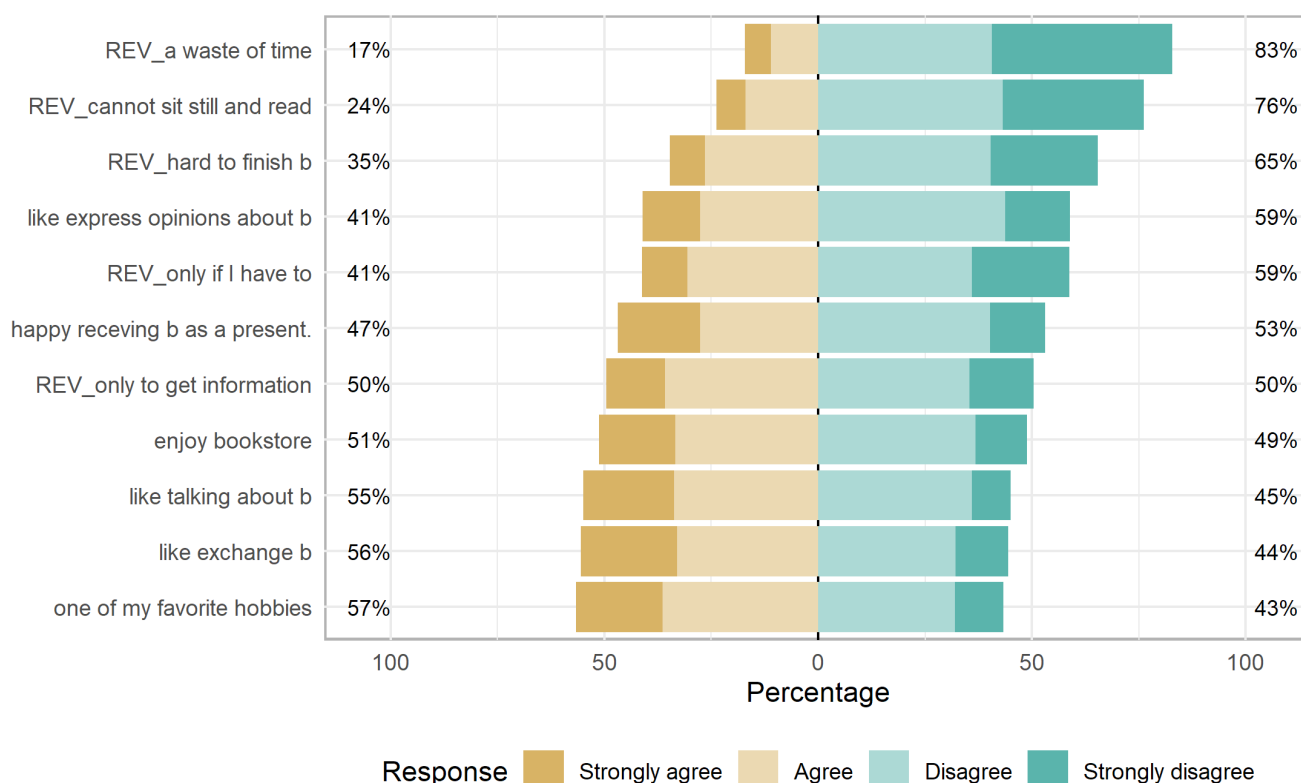


Fig. 2: Inversione dei soli punteggi: la scala nel grafico non viene interpretata correttamente

Per invertire in questo modo la polarità di tutti e quattro gli items:

```
items24 %>%
  # invertiamo la codifica
  mutate(across(.cols=c(1,4,6,8,9),
    ~fct_rev(.x)))
```

Opera in questo modo anche la funzione `reverse.levels()` del pacchetto *likert*.

Invertire i livelli e etichette

Si cambiano i valori numerici degli item, e poi si riassociano le “vecchie” etichette di livello ai rispettivi valori numerici.

Il disaccordo diventa accordo (correttamente), e torna associato ai valori 1 e 2, come per gli altri item della scala.

- **Pro:** etichette e valori dei livelli sono omogenei per tutti gli items; le analisi qualitative vengono facilitate.
- **Contro:** la rappresentazione non corrisponde più alle risposte registrate, ed è necessario rinominare la variabile con un prefisso (ad esempio: REV_x), per evitare confusione nell'interpretazione (vedi oltre).

Il risultato è come nell'immagine che segue:

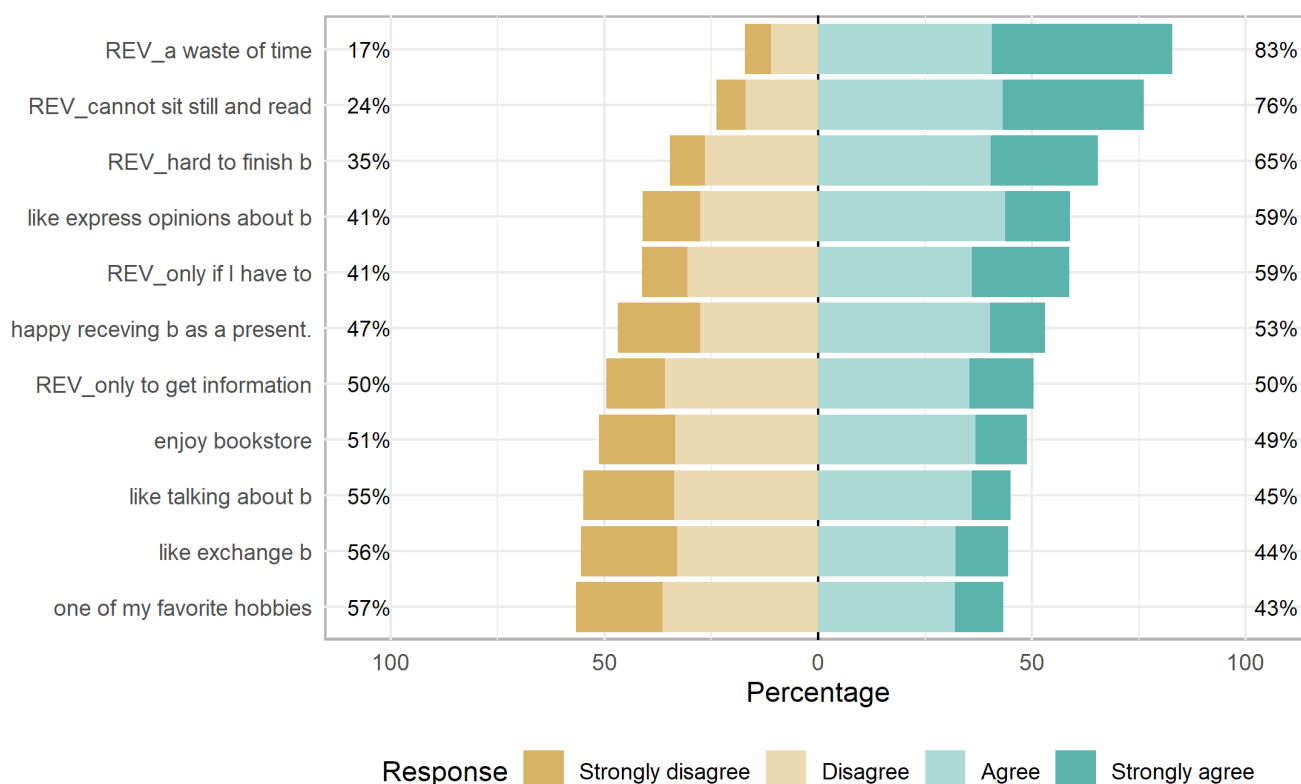


Fig. 3: Inversione dei punteggi e delle etichette

Funzione

Per semplificare le operazioni, propongo la seguente funzione, che usa `fct_rev(x)` e `fct_relabel(x, ~ levs)`, dove `levs` è il vettore dei livelli originali.

`rev_level.R`

```
rev_level <- function(x) {
  levs <- levels(x)
  x <- fct_rev(x)
  x <- fct_relabel(x, ~ levs)
  return(x)
}
```

}

[su GitHub, con documentazione](#)

Esempio:

```
# item da modificare
```

```
table(items24$`only if I have to`)
```

Strongly disagree	Disagree	Agree	Strongly agree
14947	23515	20000	7029

```
# item modificato
```

```
table(rev_level(items24$`only if I have to`))
```

Strongly disagree	Disagree	Agree	Strongly agree
7029	20000	23515	14947

Rinominare gli items

Per evitare confusione in fase di interpretazione dei dati e dei grafici, è bene rinominare gli item modificati: ad esempio, rinominare `only if I have to` in `REV_only if I have to`

```
# item modificato
```

```
table(items24$`REV_only if I have to`)
```

Strongly disagree	Disagree	Agree	Strongly agree
7029	20000	23515	14947

Per modificare in gruppo gli items

```
# intera scala
```

```
items %>%
```

```
  # invertiamo la codifica con la funzione rev_level per gli items 1, 4, 6, 8, 9
```

```
  mutate(across(.cols=c(1, 4, 6, 8, 9),
                    ~rev_level(.x))) %>%
```

Last
update:
16/09/2025 21:27 r:gestione_dei_dati:likert_inversione_polarita https://www.agnesevardanega.eu/wiki/r/gestione_dei_dati/likert_inversione_polarita

```
# cambiamo il nome degli stessi items  
rename_with(~ paste0("REV_", .x, recycle0 = TRUE),  
             c(1,4,6,8,9))
```

[Gestione dei dati](#), [Analisi monovariata](#), [Scale di atteggiamento](#)

From:

<https://www.agnesevardanega.eu/wiki/> - **Ricerca Sociale con R**

Permanent link:

https://www.agnesevardanega.eu/wiki/r/gestione_dei_dati/likert_inversione_polarita

Last update: **16/09/2025 21:27**

