

sobol Sensitivity Analysis

Fosco Mattia Vesely

Indice

1	Introduzione	2
2	Installazione	2
3	Utilizzo	2
3.1	Acquisizione dei parametri	3
3.1.1	Concludere l'introduzione dei parametri	3
3.2	Analisi Sensitività	4

1 Introduzione

Il pacchetto R `sobSenAn` è disegnato per fornire un interfaccia utente interattiva che accompagni l'utilizzatore attraverso i vari passaggi connessi alla stima degli indici di sensitività di un modello ai dati di input. Soluzioni simili sono presenti in **R** ma richiedono che il modello sia richiamabile dall'interno dello stesso ambiente **R**.

Il presente documento è destinato ad illustrare all'utente la modalità d'uso del programma. Per chi intendesse vagliare gli aspetti più tecnici si rimanda alla documentazione del pacchetto, disponibile su GitHub.

2 Installazione

Disponendo di un ambiente R già installato¹ è possibile installare `sobSenAn` con i comandi forniti in listing 1.

```
install.packages("remotes") # nel caso non sia già installato  
library(remotes)  
remotes::install_github(FoscoV/sobSenAn)
```

Listing 1: Procedura d'installazione entro R

Le istruzioni contenute in listing 1 provvedono automaticamente all'installazione della versione più recente di `sobSenAn` (od al suo aggiornamento) e delle relative dipendenze. Il pacchetto `SobSenAn` è basato su strumenti di analisi dati forniti in altri pacchetti. La struttura del pacchetto anziché uno script è stata adottata, oltre che per la semplificazione nella gestione delle versioni in vista di futuri aggiornamenti, per la più comoda gestione delle dipendenze.

3 Utilizzo

Il funzionamento del pacchetto è divisibile in parti successive:

1. acquisizione di ciascun parametro (3.1)
2. generazione dei valori di input per il modello d'interesse (3.1.1)
3. esecuzione su un modello esterno delle simulazioni secondo i parametri indicati
4. stima degli indici di sensibilità sulla base dei risultati ottenuti (3.2)

Pacchetti **R** disponibili per la stima degli indici di sensitività richiedono la disponibilità di una funzione di classe **predict** per l'iterazione di MonteCarlo sui parametri di input. La strutturazione adottata da `sobSenAn` genera i dati di input per il modello che li dovrà poi elaborare.

La fase di stima degli indici è quindi asincrona all'esecuzione del modello.

¹per l'installazione si può fare riferimento al sito ufficiale del progetto CRAN

3.1 Acquisizione dei parametri

sobSenAn contiene l'istruzione `biblio2parameter()` che avvia una procedura guidata per l'inserimento dei parametri rinvenuti in bibliografia e di seguito la relativa distribuzione.

Dapprima è richiesto il nome del parametro la cui distribuzione si intende valutare. Il nome può essere liberamente attribuito ma non deve essere "Dummy" nè iniziare con una cifra.

Vengono di seguito richiesti i valori disponibili per il parametro².

Premendo "invio" su una linea vuota si conclude la fase di introduzione dei valori e viene richiesto di scegliere la distribuzione più idonea. La scelta è presentata corredata da:

Goodness of Fit (Kolmogorov-Smirnov) che indica quanto la distribuzione corrisponde ai dati indicati

Single Effect che stima l'effetto (medio e massimo) di ciascuno dei parametri forniti nel determinare la distribuzione. Sulla base di questo è stimato quanto un ulteriore parametro possa snaturare la distribuzione (o quanto averne trovato uno di meno avrebbe cambiato l'esito dell'analisi).

Grafico un istogramma riporta la distribuzione fornita e le linee colorate ciascuna delle distribuzioni ad essa associabili.

Per completezza esplorativa, sono supportate (quindi mostrate) anche distribuzioni che inverosimilmente saranno riscontrate. I criteri su cui fondare la scelta della distribuzione più adatta richiedono l'interpretazione dell'utilizzatore.

Scelta la distribuzione, l'iter automatizzato richiede alcune ulteriori informazioni. Sono supportate distribuzioni discrete, aventi cioè numeri interi³, qualora sia il caso, rispondere positivamente a questa richiesta.

Distribuzioni troncate possono essere definite (se richiesto) indicando i valori limite delle probabilità associate. Canonicamente le code inferiori alla probabilità 0.1 e superiori a 0.9. Alternativamente i punti di troncatura possono essere indicati per il loro valore assoluto: alcuni parametri possono avere un limite inferiore pari a 0 (non accettando valori nulli e negativi) e privi di un limite superiore.

In virtù di una sensibilità di eFAST ai valori estremi della distribuzione dei parametri, sono impiegate le troncature di default a 0.1 e 0.9 della curva di distribuzione di probabilità⁴.

3.1.1 Concludere l'introduzione dei parametri

Rispondendo "n" alla domanda se inserire un ulteriore parametro, si accede all'ultima fase prima dell'esecuzione dei modelli. Su indicazione dell'utilizzatore è generato un file contenente una colonna per ciascuno dei parametri introdotti oltre ad una colonna "Dummy" aggiuntiva (requisito per in confronto tra i parametri eseguito entro il programma). Il file dovrà essere impiegato come input al modello di simulazione. I risultati

²possono essere inseriti uno alla volta premendo "invio" una volta dopo ciascun inserimento oppure intervallare i valori con " , "

³Un esempio potrebbero essere livelli di resistenza ad un patogeno

⁴si ricordi che una proprietà delle curve di distribuzione è che una probabilità non nulla sia associata a tutti i valori compresi tra $-\infty$ e $+\infty$

andranno aggiunti in una colonna di modo tale che corrispondano alla riga di input con cui sono stati ottenuti.

Possono essere inserite diverse colonne di risultati che saranno tutte analizzate in funzione dei dati di input (interazioni tra output di simulazioni non sono supportate).

3.2 Analisi Sensitività

`output2sens()` provvede autonomamente a tutte le parti di analisi. Richiede di indicare il file contenente gli input (e gli output), forniti (ed ottenuti) (d)al modello di simulazione.

Viene chiesto di indicare un file che sarà popolato con:

1. `SAResults.csv` contenente gli indici per ciascun parametro
2. un file `.pdf` con i grafici relativi a ciascun risultato fornito dal modello