# Successive Choice

T.C.A Madhav Raghavan

January 25, 2011

**Abstract**

We explore the case where an agent encounters alternatives in the form of a list. We characterise a successive choice function over lists in the discrete case. We explore extensions of this characterisation to the probabilistic case.

## 1 Introduction

In some contexts, the structure of the choice problem is of relevance to the actual choice made. For example, when selecting from a list of applicants, or scrolling down the Google search result page, or even picking the next job to apply for during a career, we find that alternatives are not presented to the agent all at once, i.e., in set form. Instead, the agent encounters alternatives in the form of a list and must evaluate them sequentially.

While we are at first interested in this sequential choice from lists, we are also keen to explore notions of path dependence, which is the idea that different arrangements of alternatives can lead to different outcomes.

There are many different ways to choose from a list (Rubinstein, Salant 2006). However, we find that successive choice is of particular interest. Most importantly, this is because it carries a sequential logic.

## 2 The Basic Model

We consider a finite set of alternatives X. A **list over X** is a finite sequence of non-repeating alternatives $(a_1, \ldots, a_k)$ from X. We denote $\mathcal{L}$ as the set of all lists over X.

The concatenation of two lists $L_1$ and $L_2$ is written as $< L_1, L_2 >$.

A **choice function from lists** is a function $C : \mathcal{L} \to X$ such that for every $L \in \mathcal{L}$, $C(L) = a_j$, for some $a_j \in L$.

Let $(a_1, \quad a_2, \quad a_3, \quad \ldots, \quad a_n)$ be a list. A **Successive Choice Function (SCF)** evaluates alternatives in a sequential fashion as described below.

- The first element is selected. $d_1 \equiv a_1$

- The SCF then compares the first two alternatives and selects one. $d_2 \equiv$ t($d_1$,$a_2$)
  ...
- At stage $j$, the thus-far selected alternative is compared to alternative j. $d_j \equiv$ t($d_{j-1}$,$a_j$)
  ...
- The last item is compared. $d_n \equiv$ t($d_{n-1}$, $a_n$)
- The item that survives is the chosen alternative. So $C^S(a_1,\ldots,a_n) = d_n$

So if $C^S$ is an SCF, then $C^S$(L)= t(t(...t(t($a_1, a_2$), $a_3$) ...), $a_n$).

Rubinstein and Salant describe an axiom for choice functions from lists. A choice function from lists C satisfies **Partition Independence** if for all lists L and all disjoint partitions $L_1$ and $L_2$ of L:

$$\mathrm{C(L)} = \mathrm{C(C(}L_1\mathrm{),C(}L_2\mathrm{))}$$

PI is too strong a condition for an SCF. That is, every choice function that satisfies PI is an SCF, but an SCF may fail to satisfy PI.

## 3 Two Axioms

In this section we propose two axioms for choice functions from lists. We show that they are independent, and that collectively they are weaker than PI.

Definition **1** *A choice function from lists C satisfies* **Right Uniformity (RU)** *if for any list L=($a_1,\ldots,a_n$) such that C(L)=$a_j$, and for any list $L_1$ over $\{a_{j+1},\ldots,a_n\}$,*

$$C(< a_j, L_1 >) = a_j.$$

Definition **2** *A choice function from lists C satisfies* **Left Consistency (LC)** *if for any list L=($a_1,\ldots,a_n$) such that C(L)=$a_j$,*

$$C(C(a_1,\ldots,a_{j-1}), a_j) = a_j.$$

Claim **1** *RU and LC are independent.*

*Proof*: The symbol in boldface refers to the chosen alternative in each case.

Suppose C($b,\mathbf{c}$), C($a,\mathbf{b}$), but C($a,\mathbf{b},c$). Here RU is violated, but LC is satisfied.

Suppose, on the other hand, that C($\mathbf{b},c$), C($\mathbf{a},b$), C($\mathbf{a},c$), but that C($a,\mathbf{b},c$). In this case, RU is satisfied but LC is violated. ■

As discussed, PI is a stronger condition than RU and LC.

Claim **2** *PI implies RU and LC, but RU and LC do not imply PI.*

*Proof*: Let C satisfy PI. Let L=$(a, b, c)$. Suppose C$(a, b, c) = a$. LC is trivially satisfied. By PI, C(C$(a, b)$,$c$)=$a$ $\Rightarrow$ C$(a, b)$=$a$ and C$(a, c)$=$a$. So RU is satisfied.
Instead, suppose C$(a, b, c) = b$. By PI, C(C$(a, b)$,$c$)=$b$ $\Rightarrow$ C$(a, b)$=$b$ and C$(b, c)$=$b$. So LC and RU are satisfied.
Now, suppose C$(a, b, c) = c$. RU is trivially satisfied. By PI, C(C$(a, b)$,$c$)=$c$, and so LC is satisfied.
So RU and LC are satisfied whenever PI is satisfied.

However, RU and LC are weaker conditions than PI.

Suppose we have C$(\mathbf{b}, c)$, C$(\mathbf{a}, b)$, and C$(a, \mathbf{c})$, but C$(a, b, \mathbf{c})$. Here, RU is trivially satisfied. Since C(C$(a, b), \mathbf{c}$), LC is satisfied. And because C$(\mathbf{a}, C(b, c))$, PI is violated. ■

## 4 Characterisation

Proposition **1** *A choice function from lists C is an SCF if and only if it satisfies RU and LC.*

*Proof*: We first show that an SCF satisfies RU and LC.
Let $C^S$ be an SCF, and let $C^S(a_1, \ldots, a_n) = a_j$.
$\Rightarrow t(...t(t(a_1, a_2), a_3)...a_n) = a_j$.
$\Rightarrow$ In the $j^{th}$ iteration, $C^S(C^S(a_1, \ldots, a_{j-1}), a_j) = a_j$, so LC is satisfied.
Also, $t(a_j, a_i) = a_j \ \forall \ i > j$.
Let L* be any list over $\{a_{j+1}, \ldots, a_n\}$.
It is easy to show that $C^S(< a_j, L* >) = a_j$. So RU is satisfied.

In the reverse direction, we proceed in two steps. We show first for a three-element list that any choice function that satisfies RU and LC must be an SCF. We then extend this result to an arbitrary list of n elements using an induction argument.

So Let L be a list of 3 elements. Let C satisfy RU and LC, and let L=$(a, b, c)$. We have to show C$(a, b, c)$ = C(C$(a, b), c$).
**Case 1: C$(a, b, c)$=$a$)**
From RU, we know that C$(a, b)$=$a$ and C$(a, c)$=$a$. So C(C$(a, b), c$)=$a$.
**Case 2: C$(a, b, c)$=$b$)**
From RU, we know that C$(b, c)$=$b$. From LC, we know that C$(a, b)$=$b$. So C(C$(a, b), c$)=$b$.
**Case 3: C$(a, b, c)$=$c$)**
From LC, we directly know that C(C$(a, b), c$)=$c$.
So C is an SCF. ■

We now prove the following.

Lemma 1 *Let C be an SCF for a list with n elements. If C satisfies RU and LC, then C is an SCF for any right-extension of the list to $n+1$ elements.*

*Proof*: Suppose C$(a_1, \ldots, a_n)$ = C$(\ldots(C(a_1, a_2), a_3) \ldots a_n) = a_j$.

To show:

$$C(a_1, \ldots, a_{n+1}) = \begin{cases} a_j & : C(a_j, a_{n+1}) = a_j \\ a_{n+1} & : C(a_j, a_{n+1}) = a_{n+1} \end{cases}$$

Case 1: C$(a_j, a_{n+1})$=$a_j$ Suppose C$(a_1, \ldots, a_{n+1}) = a_k \neq a_j$.
By LC, C(C$(a_1, \ldots, a_{k-1}), a_k) = a_k$. By RU, C$(a_k, \ldots, a_n)$=$a_k$.
C is an SCF over $(a_1, \ldots, a_n) \Rightarrow$ C$(a_1, \ldots, a_n)$=$a_k$.
Contradiction. So C$(a_1, \ldots, a_{n+1}) = a_j$.

Case 2: C$(a_j, a_{n+1})$=$a_{n+1}$ Suppose C$(a_1, \ldots, a_{n+1}) = a_k \neq a_{n+1}$.
By LC, C(C$(a_1, \ldots, a_{k-1}), a_k) = a_k$. By RU, C$(a_k, \ldots, a_n)$=$a_k$.
C is an SCF over $(a_1, \ldots, a_n) \Rightarrow$ C$(a_1, \ldots, a_n)$=$a_k$.
Contradiction. So C$(a_1, \ldots, a_{n+1}) = a_{n+1}$. ■

So we have shown that any choice function that satisfies RU and LC must be an SCF. This completes the characterisation.

## 5 Probabilistic Choice

A Probabilistic Choice Function C from lists assigns to every list L=$(a_1, \ldots, a_n)$ a probability distribution $(\alpha_1, \ldots, \alpha_n)$, such that $\alpha_i$ is the probability that C picks $a_i$ from L.
Naturally, $\Sigma_i \alpha_i = 1$.
In the deterministic case, this produces a vector of 0s and one instance of 1, which identifies the chosen element.

We illustrate the working of a Probabilistic Successive Choice Function (PSCF) with an example. Suppose we have the list:

$$((a, \quad b), \quad c)$$

- Suppose the probabilities attached to $(a, b)$ are $(p, 1-p)$, to $(b, c)$ are $(q, 1-q)$, and to $(a, c)$ are $(r, 1-r)$.
- Then $\alpha_a(a, b, c) = pr$
- $\alpha_b(a, b, c) = (1-p)q$
- $\alpha_c(a, b, c) = [p(1-r) + (1-p)(1-q)]$
- e.g., let $p = 0.8, q = 0.6, r = 0.3$
- Then $\alpha_a(a, b, c) = 0.24, \alpha_b(a, b, c) = 0.12, \alpha_c(a, b, c) = 0.64$.

Definition 3 *A choice function from lists C satisfies* **Probabilistic Right Uniformity (PRU)** *if for any list L=(*$a_1,\ldots,a_n$*),* $\forall a_j \in L$ *such that* $\alpha_j^C(L) \neq 0$*, and for any list* $L_1$ *over* $\{a_{j+1},\ldots,a_n\}$,

$$\alpha_j^C(< a_j, L_1 >) \neq 0.$$

Definition 4 *A choice function from lists C satisfies* **Probabilistic Left Consistency (PLC)** *if for any list L=(*$a_1,\ldots,a_n$*), for all* $a_j \in L$ *such that* $\alpha_j^C(L) \neq 0$, $\exists a_k \in \{a_1,\ldots,a_{j-1}\}$ *such that*

$$\alpha_k^C(a_1,\ldots,a_{j-1})) \neq 0 \text{ and } \alpha_j^C(a_k,a_j) \neq 0.$$

We make the following observations:

- The axioms PRU and PLC reduce to RU and LC when $\alpha$ is allowed to take on values of 0 and 1 only (discrete case).
- PRU and PLC are both necessary for a PCF to be successive in nature.
- But they are not sufficient, i.e., while they guarantee that a choice function will work successively, they are not strong enough to guarantee the same probability distribution over the list.