Ruby - Feature #3647

Array#sample(n, replace=false)

08/03/2010 01:13 PM - mrkn (Kenta Murata)

Status:	Feedback	
Priority:	Normal	
Assignee:	mrkn (Kenta Murata)	
Target version:		

Description

Array#sample does not support sampling with replacement directly, so we must write codes like the following to obtain samples with replacement.

```
samples_with_replacement = n.times.map { source.sample }
```

If Array#sample has the argument for switching with/without replacement, we can write it as follow.

```
samples_with_replacement = source.sample(n, true)
```

The patch was attached. Please consider to merge.

Related issues:

Related to Ruby - Feature #3649: Array#sample, Array#shuffle, Array#shuffle! ...	**Closed**	**08/03/2010**
Related to Ruby - Feature #4247: New features for Array#sample, Array#choice	**Rejected**	

History

#1 - 08/03/2010 01:45 PM - mrkn (Kenta Murata)

むらたです。

dev に英語で投稿してしまいました。ごめんなさい。
無理して英語で書いて投稿先を core にしていたと思っていたのですが・・・

On 2010/08/03, at 13:13, Kenta Murata wrote:

> Feature #3647: Array#sample(n, replace=false)
> http://redmine.ruby-lang.org/issues/show/3647
>
> 起票者: Kenta Murata
> ステータス: Open, 優先度: Normal
> カテゴリ: core, Target version: 1.9.x
>
> Array#sample does not support sampling with replacement directly, so we must write codes like the following to obtain samples with replacement.
>
> samples_with_replacement = n.times.map { source.sample }
>
> If Array#sample has the argument for switching with/without replacement, we can write it as follow.
>
> samples_with_replacement = source.sample(n, true)

この提案は、Array#sample メソッドで復元抽出もできるようにしたいという事です。
同様のインタフェースを持つ例として R の sample 関数があります。

--
Kenta Murata
OpenPGP FP = 1D69 ADDE 081C 9CC2 2E54 98C1 CEFE 8AFB 6081 B062

本を書きました!!
『Ruby 逆引きレシピ』 http://www.amazon.co.jp/dp/4798119881/mrkn-22

E-mail: mrkn@mrkn.jp
twitter: http://twitter.com/mrkn/
blog: http://d.hatena.ne.jp/mrkn/

#2 - 08/03/2010 04:53 PM - mrkn (Kenta Murata)

- File array_sample_with_replace_hash.patch added

replace の指定を、オプショナルなハッシュを経由して行うように patch を変更しました。
こうすると、Array#sample に対してオプショナルな引数を追加しやすくなりますよね。
例えば、乱数生成器を外部から指定するとか。

#3 - 08/03/2010 06:25 PM - mrkn (Kenta Murata)

- File array_sample_shuffle.patch added

IRC でなかださんが random 引数にも対応した patch を作ってくださいました。
http://www.atdot.net/sp/raw/0dgk6l

Array#sample がデフォルトでない乱数生成器を受け付けるならば、
Array#shuffle と Array#shuffle! も同じように受け付けないと不自然だと思います。
そこで、中田さんの patch をベースに Array#shuffle と Array#shuffle! に対して random 引数を追加する変更をし、
せっかくなのでテストも追記した新しい patch を作成しました。

#4 - 08/03/2010 08:27 PM - mrkn (Kenta Murata)

先程、乱数生成器を指定する変更も含んだ patch を投稿しましたが、
もともとは Array#sample を復元抽出に対応させるための feature request でした。

そこで、乱数生成器の指定に関する変更点は別の feature request として分離することにしました。

ということで、この feature request の最終的な提案 patch は #2 で追加したファイルであると考えて下さい。

#5 - 12/10/2010 11:40 PM - tarui (Masaya Tarui)

Array#sampleの提案が出ていて思い出しました。
これはどうなってるんでしたっけ?

> replace の指定を、オプショナルなハッシュを経由して行うように patch を変更しました。
> こうすると、Array#sample に対してオプショナルな引数を追加しやすくなりますよね。
> 例えば、乱数生成器を外部から指定するとか。

オプショナルなハッシュ経由は名前を覚えなければならないという点で反対したいです。

サンプリングで再抽選(復元抽出?)するかどうかはかなり基本的な話だと思うので、
オプションをつけること自体は賛成します。

#6 - 12/11/2010 08:56 AM - matz (Yukihiro Matsumoto)

まつもと ゆきひろです

In message "Re: [[ruby-dev:42743]]() [Ruby 1.9-Feature#3647] Array#sample(n, replace=false)"
on Fri, 10 Dec 2010 23:40:21 +0900, Masaya Tarui [redmine@ruby-lang.org]() writes:

> Array#sampleの提案が出ていて思い出しました。
> これはどうなってるんでしたっけ?

合意がとれず、放置だと思います。

>> replace の指定を、オプショナルなハッシュを経由して行うように patch を変更しました。
>> こうすると、Array#sample に対してオプショナルな引数を追加しやすくなりますよね。
>> 例えば、乱数生成器を外部から指定するとか。
>
> オプショナルなハッシュ経由は名前を覚えなければならないという点で反対したいです。
>
> サンプリングで再抽選(復元抽出?)するかどうかはかなり基本的な話だと思うので、
> オプションをつけること自体は賛成します。

キーワード引数には反対だが、オプションには賛成であるとなると、
具体的にはどのような「オプション」を想定しておられますか?

#7 - 12/11/2010 11:05 PM - tarui (Masaya Tarui)

たるいです。

2010年12月11日8:55 Yukihiro Matsumoto [matz@ruby-lang.org]():

> まつもと ゆきひろです
>
> In message "Re: [ruby-dev:42743] [Ruby 1.9-Feature#3647] Array#sample(n, replace=false)"
> on Fri, 10 Dec 2010 23:40:21 +0900, Masaya Tarui redmine@ruby-lang.org writes:
>
>> Array#sampleの提案が出ていて思い出しました。
>> これはどうなってるんでしたっけ?
>
> 合意がとれず、放置だと思います。

失礼ながら特に議論した形跡が分からなかったんですが、
どこかに合意が取れない項目があったんでしょうか?

>>> replace の指定を、オプショナルなハッシュを経由して行うように patch を変更しました。
>>> こうすると、Array#sample に対してオプショナルな引数を追加しやすくなりますよね。
>>> 例えば、乱数生成器を外部から指定するとか。
>>
>> オプショナルなハッシュ経由は名前を覚えなければならないという点で反対したいです。
>>
>> サンプリングで再抽選(復元抽出?)するかどうかはかなり基本的な話だと思うので、
>> オプションをつけること自体は賛成します。
>
> キーワード引数には反対だが、オプションには賛成であるとなると、
> 具体的にはどのような「オプション」を想定しておられますか?

何もないよりはキーワード引数でもあった方がいいとは思っていますが、
想定してるものは、最初のむらけんさんの提案の方です。

```
[1,2,3,4].sample   #=>1個なので再抽選関係なし
[1,2,3,4].sample(3)  #=>再抽選なし
```

ここまでが現在の仕様ですが、

```
[1,2,3,4].sample(3,true)  #=>再抽選あり
[1,2,3,4].sample(3,false)  #=>再抽選なし
```

が追加になります。

以上、ご検討お願いします。

--
樽家昌也(Masaya TARUI)
No Tool,No Life.

#8 - 12/11/2010 11:38 PM - matz (Yukihiro Matsumoto)

まつもと ゆきひろです

In message "Re: [ruby-dev:42751] Re: [Ruby 1.9-Feature#3647] Array#sample(n, replace=false)"
on Sat, 11 Dec 2010 23:04:11 +0900, Masaya TARUI tarui@prx.jp writes:

> 失礼ながら特に議論した形跡が分からなかったんですが、
> どこかに合意が取れない項目があったんでしょうか?

「本当に必要なの」ってところがですかね。

> 何もないよりはキーワード引数でもあった方がいいとは思っていますが、
> 想定してるものは、最初のむらけんさんの提案の方です。
>
> ```
> [1,2,3,4].sample   #=>1個なので再抽選関係なし
> [1,2,3,4].sample(3)  #=>再抽選なし
> ```
>
> ここまでが現在の仕様ですが、
>
> ```
> [1,2,3,4].sample(3,true)  #=>再抽選あり
> [1,2,3,4].sample(3,false)  #=>再抽選なし
> ```
>
> が追加になります。

反対します。trueの意味がなんであるか、絶対にわからなくなるか
らです。たとえば現在でも instance_methods は真偽値をオプショ
ンとして受け付けますが、trueの時にどう振舞うかは私でさえ覚え
ていられません。これは失敗だったと思っています。

#9 - 12/19/2010 11:22 PM - mame (Yusuke Endoh)

遠藤です。

2010年12月11日23:38 Yukihiro Matsumoto matz@ruby-lang.org:

> まつもと ゆきひろです
>
> In message "Re: [ruby-dev:42751] Re: [Ruby 1.9-Feature#3647] Array#sample(n, replace=false)"
> ? ?on Sat, 11 Dec 2010 23:04:11 +0900, Masaya TARUI tarui@prx.jp writes:
>
> > 失礼ながら特に議論した形跡が分からなかったんですが、
> > どこかに合意が取れない項目があったんでしょうか?
>
> 「本当に必要なの」ってところがですかね。

同じく、この機能は

```
(1..n).map { ary.sample }
```

で済むので、

```
ary.sample(replace: true)
```

じゃ字数も変わらないし、本当にいるのかなという気がします。
まあ Array#sample 自体も ary[rand(ary.size)] で済むといえば
済むのですが。

> > 何もないよりはキーワード引数でもあった方がいいとは思っていますが、
> > 想定してるものは、最初のむらけんさんの提案の方です。
> >
> > ```
> > [1,2,3,4].sample ?#=>1個なので再抽選関係なし
> > [1,2,3,4].sample(3) #=>再抽選なし
> > ```
> >
> > ここまでが現在の仕様ですが、
> >
> > ```
> > [1,2,3,4].sample(3,true) #=>再抽選あり
> > [1,2,3,4].sample(3,false) #=>再抽選なし
> > ```
> >
> > が追加になります。
>
> 反対します。trueの意味がなんであるか、絶対にわからなくなるか
> らです。たとえば現在でも instance_methods は真偽値をオプショ
> ンとして受け付けますが、trueの時にどう振舞うかは私でさえ覚え
> ていられません。これは失敗だったと思っています。

まつもとさんと同じでこれには反対です。

入れるとしたら sample(何かキーワード: true) だと思うんですが、
元の提案の "replace" は、非ネイティブかつ統計用語に詳しくない
私としては、選択された要素を破壊的に取り除くように思えてしまう
(つまり以下の挙動を期待する) ので嫌らしいです。

```
ary = [1, 2, 3]
p ary.sample(replace: true) #=> 1
p ary #=> [2, 3]
```

よく考えるとこの挙動は別に replace してないんですが、replace
には破壊的な語感がありすぎてこんなふうに思ってしまいます。

余談ですが、この挙動はこの挙動で欲しい。

duplication: true とかどうなのかなあ。ruby-core に持っていくと
いいかもしれないですね。英語圏だと replacement で完全制覇されて

いるという噂もありますが。

--
Yusuke Endoh mame@tsg.ne.jp

#10 - 12/20/2010 11:09 PM - akr (Akira Tanaka)

2010年12月19日23:22 Yusuke ENDOH mame@tsg.ne.jp:

> 入れるとしたら sample(何かキーワード: true) だと思うんですが、
> 元の提案の "replace" は、非ネイティブかつ統計用語に詳しくない
> 私としては、選択された要素を破壊的に取り除くように思えてしまう
> (つまり以下の挙動を期待する) ので嫌らしいです。
>
> ```
> ary = [1, 2, 3]
> p ary.sample(replace: true) #=> 1
> p ary #=> [2, 3]
> ```
>
> よく考えるとこの挙動は別に replace してないんですが、replace
> には破壊的な語感がありすぎてこんなふうに思ってしまいます。
>
> 余談ですが、この挙動はこの挙動で欲しい。
>
> duplication: true とかどうなのかなあ。ruby-core に持っていくと
> いいかもしれないですね。英語圏だと replacement で完全制覇されて
> いるという噂もありますが。

unique とか。
現在の挙動 (非復元抽出) が unique: true で。
unique: false とすると復元抽出。

(uniq でも)

--
[田中 哲][たなか あきら][Tanaka Akira]

#11 - 03/31/2012 10:17 AM - mame (Yusuke Endoh)

- Description updated

- Status changed from Open to Feedback

- Assignee set to mrkn (Kenta Murata)

mrkn さん、

matz の「本当に必要なの」というのに返事と、キーワードの名前の候補出しをお願いします。

--
Yusuke Endoh mame@tsg.ne.jp

#12 - 10/25/2012 06:03 PM - yhara (Yutaka HARA)

- Target version changed from 2.0.0 to 2.6

#13 - 12/25/2017 06:14 PM - naruse (Yui NARUSE)

- Target version deleted (2.6)

#14 - 03/10/2020 04:58 AM - nobu (Nobuyoshi Nakada)

- Description updated

https://github.com/ruby/ruby/pull/2955
The option name is replace for now.

#15 - 03/10/2020 10:18 AM - sawa (Tsuyoshi Sawada)

Array#repeated_combination, Array#repeated_permutation が既にあり、本件はこれと類似した内容なので、一貫性の観点から Array#repeated_sample という名の独立したメソッドが良いと思います。あるいは、それが数学用語として間違っているなら、 Array#replaced_sample でよいかなという気もします。

#16 - 03/11/2020 02:09 AM - nobu (Nobuyoshi Nakada)

https://github.com/ruby/ruby/pull/2956

Files

array_sample_with_replace.patch	3.64 KB	08/03/2010	mrkn (Kenta Murata)
array_sample_with_replace_hash.patch	4.35 KB	08/03/2010	mrkn (Kenta Murata)
array_sample_shuffle.patch	9.62 KB	08/03/2010	mrkn (Kenta Murata)