
dedup

syntax: `dedup(vec)`

purpose: Removes any repeated values from the input vector, returning a new vector that has just one copy of each distinct value in the input.

examples:

```
>> x = [1 1 1 1 1 1 1 1];
>> dedup(x) => ans: 1
>> y = [1 1 1 1 1 1 1 2];
>> dedup(y) => ans: [1 2]
```

You are part of an ambitious project to find risk factors for hepatitis A, which can be transmitted by casual contact such as shaking hands. An important question is whether it is the number of exposures to infected people, including repeats, or the number of distinct infected individuals contacted. Through some feat of magic — seen only in movies or statistics textbooks — you have a list for each subject of each of the contacts with individuals with hepatitis A. Each infected person has an identification number which you use to code the data.

```
>> bill = [79 83 64 128 79 246];
>> monica = [83 83 83 83 83];
```

Bill has contacts with many people with few repeats. Monica works in a small office and has regular contact with a single infected person. The total number of contacts is:

```
>> length(bill) => ans: 6
```

```
>> length(monica) => ans: 5
```

The total number of contacted infected individuals is

```
>> length( dedup(bill) ) => ans: 5
```

```
>> length( dedup(monica) ) => ans: 1
```

See also: MULTIPLES

This document is an excerpt from

Resampling Stats in MATLAB

Daniel T. Kaplan

Copyright (c) 1999 by Daniel T. Kaplan, All Rights Reserved

This document differs from the published book in pagination and in the omission (unintentional, but unavoidable for technical reasons) of figures and cross-references from the book. It is provided as a courtesy to those who wish to examine the book, but not intended as a replacement for the published book, which is available from

Resampling Stats, Inc.

www.resample.com

703-522-2713

