# Why am I losing precision with large whole numbers in Stata, such as an ID variable?

Authored by

**stats writer**

June 30, 2024

When working with large whole numbers in Stata, such as ID variables, it is possible to experience a loss of precision. This is due to the fact that Stata stores all numbers as double-precision floating-point numbers, which have a limited number of digits they can accurately represent. As the numbers get larger, the available digits for precision decrease, resulting in potential rounding errors and loss of precision. Additionally, Stata may also display the numbers in scientific notation, further reducing the number of digits displayed. Therefore, it is important to be aware of this limitation when working with large whole numbers in Stata to ensure accurate results.

## Why am I losing precision with large whole numbers (such as an ID variable)? | Stata FAQ

## Say that we have a tiny data file which has just ID variables like the one below.

**input id**
**123456789**
**123456790**
**123456791**
**123456792**
**123456793**
**123456794**
**123456795**
**123456796**
**end**

## If we go to list out the values, they are displayed in scientific notation so it is hard to read the values.

**list**

**id**
**1. 1.23e+08**
**2. 1.23e+08**
**3. 1.23e+08**
**4. 1.23e+08**
**5. 1.23e+08**
**6. 1.23e+08**
**7. 1.23e+08**
**8. 1.23e+08**

**We can use the format command to tell**
**Stata that we would like it to display the values with 9 values before the decimal place,**
**and with no values after the decimal, as shown below. This way we can clearly see**
**the values for id and we can see that the ID values were not stored**
**properly.**

**format id %9.0f**
**list**

**id**

1. 123456792
2. 123456792
3. 123456792
4. 123456792
5. 123456792
6. 123456792
7. 123456792
8. 123456800

If we use the describe command, we can
see that Stata stored this value with the type float. The problem
is that a float can only store an integer value with up to 7 digits of
accuracy (but our id values were 9 digits).

describe

Contains data
obs: 8
vars: 1
size: 64 (99.9% of memory free)
---
1. id float %9.0f

---

**Sorted by:**

**Note: dataset has changed since last saved**

**If you are storing an identification number (like we are), we need our values to be stored with perfect accuracy. If your variable contains just whole numbers (like our id) variable and is up to 9 digits, you can store it as a long integer, or if it can be up to 16 digits, you can store it as a double. If your identification variable was over 16 digits long, you could store the variable as a string variable without any loss of precision (but you would not be able to do any numerical computations with it).**

**Here is an example showing how to read the variable id as a long integer.**

**input long id**
**123456789**
**123456790**

**123456791**
**123456792**
**123456793**
**123456794**
**123456795**
**123456796**
**end**

**format id %9.0f**
**list**

**id**
**1. 123456789**
**2. 123456790**
**3. 123456791**
**4. 123456792**
**5. 123456793**
**6. 123456794**
**7. 123456795**
**8. 123456796**

**Here is an example showing how to read the variable id as a string variable with a length of 9 (since the ID variable is 9).**

```
input str9 id
123456789
123456790
123456791
123456792
123456793
123456794
123456795
123456796
end

list
id
1. 123456789
2. 123456790
3. 123456791
4. 123456792
5. 123456793
6. 123456794
7. 123456795
8. 123456796
```

For more information, see the Stata

**manual or Stata Help for datatypes.**