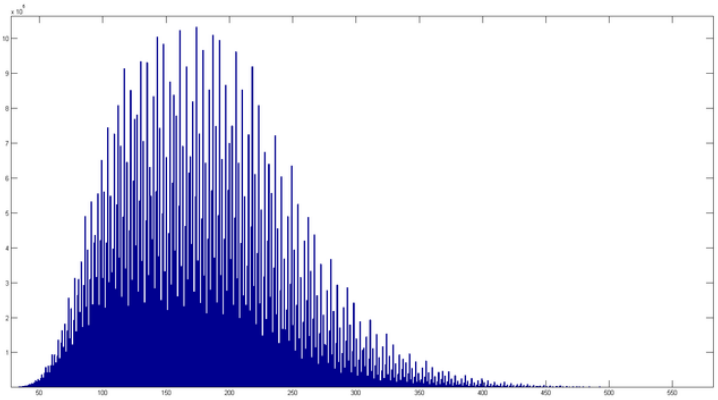
+2 join this community tour help

TeX - LaTeX Stack Exchange is a question and answer site for users of TeX, LaTeX, ConTeXt, and related typesetting systems. It's 100% free, no registration required. Join ×

How can I plot bigger amounts of data?

I am currently playing around with hailstone sequences (collatz numbers).

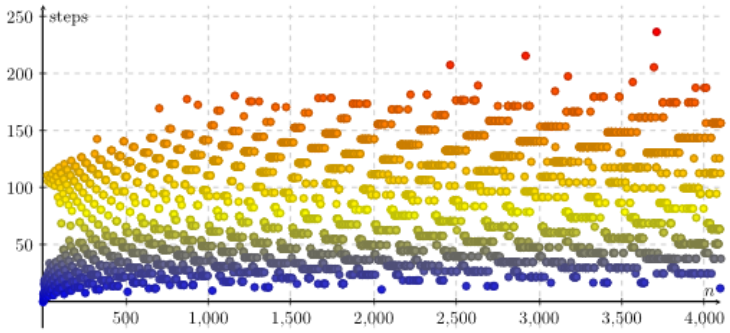
I would like to create a plot like this (from Wikipedia):



But I don't know how this kind of plot is called (it's neither a line plot, nor a scatter plot or a bar chart or a histogram, although it looks similar).

I would like to plot how many steps you need from a number `n` to get to `1`. The x-axis should be numbers (from 1 to M, with M as high as possible) and the y-axis should be the number of steps.

As I didn't know how the kind of plot above is called (and I was not able to find a name) I used a scatterplot:



Sadly, this seems to consume a lot of memory. Plotting 4100 (n, steps from n to 1) pairs worked, but when I increased it to 4150 pairs, I got:

```
! TeX capacity exceeded, sorry [main memory size=3000000].
\endvarwidth ...\vbox \bgroup \unvcopy \@tempboxa
                                                \@tempdima -\maxdimen \let...
l.28 \end{document}
```

I think the following could eventually solve my problem:

- switch to gnuplot (I have no idea how to read a CSV-file with gnuplot and I only used gnuplot+LaTeX once. Does anybody have a minimal working example with instructions how to compile it?)
- switch diagram type to the diagram above (How is it called?)
- don't actually write the numbers n in csv file, but get it from the line number (How can I get the line number from a CSV file in my code below for x?)
- switch to another type of plot like here (I guess this will be more complicated and it seems as if I had to use gnuplot / R. Does anybody know how to apply this to my problem?)

All code I've written is on GitHub. This is the LaTeX code I've written so far:

```
\documentclass[varwidth=true, border=2pt]{standalone}
\usepackage[utf8]{inputenc} % this is needed for umlauts
\usepackage[ngerman]{babel} % this is needed for umlauts
\usepackage[T1]{fontenc}    % this is needed for correct output of umlauts in pdf
\usepackage[margin=2.5cm]{geometry} %layout

\usepackage{pgfplots}

\begin{document}
\begin{tikzpicture}
    \begin{axis}[
            axis x line=middle,
            axis y line=middle,
            enlarge y limits=true,
            %xmin=0, xmax=2150,
            %ymin=0, ymax=600,
            width=15cm, height=8cm,     % size of the image
            grid = major,
            grid style={dashed, gray!30},
            ylabel=steps,
            xlabel=$n$,
            legend style={at={(0.1,-0.1)}, anchor=north}
        ]
         \addplot[scatter,only marks] table [x=n, y=steps, col sep=comma]
{../steps.csv};
    \end{axis}
\end{tikzpicture}
\end{document}
```

I would like to plot at least 100,000 elements (rather 10,000,000 if possible). How can I do it?

{pgfplots} {gnuplot} {csv}

asked May 16 '13 at 9:24

moose
6,299 3 28 98

1 Is the first one like a comb plot (see the `pgfplots` manual)? – Torbjørn T. May 16 '13 at 9:27

Yes, xcomb (pgfplots.sourceforge.net/gallery.html) looks like what I would like to get :-) I'll try it in two hours (currently I'm in a lecture). Do you know how to get rid of the dots at the end? – moose May 16 '13 at 9:42

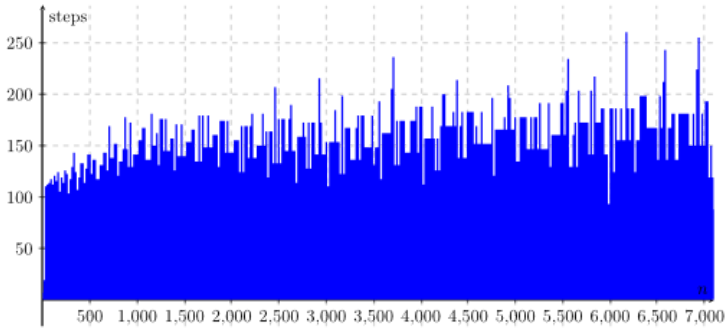
Not sure if this will help much, but you can increase the capacity of TeX's memory. See for example this. – Marc van Dongen May 16 '13 at 9:51

Torbjørn T.: Thanks, it worked as expected (I couldn't resist to try it and leaved the lecture). @MarcvanDongen: This will not help, as I have 4GB in total and I already gave TeX 3GB. – moose May 16 '13 at 10:22

2 Answers

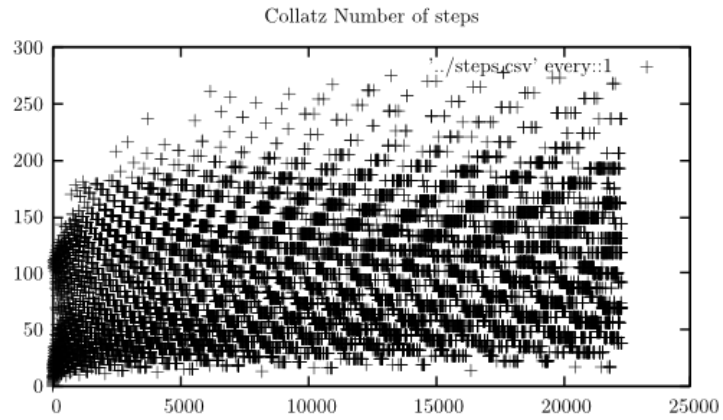
pgfplots, comb plot

Thanks to Torbjørn T., who has suggested to look at `comb plots` . This allowed me to plot 7150 (but not 7200) data points:



gnuplot

crosses



You have to create a file that tells gnuplot what to do. I've called it `plot.gnuplot` :

```
set terminal latex
set output "plot-tmp.tex"
set datafile separator ","
set title "Collatz Number of steps"
plot '../steps.csv' every::1
```

Then you have to run gnuplot: `gnuplot plot.gnuplot` This will produce a "plot-tmp.tex" that can be included into your LaTeX document:

```
\documentclass{standalone}

\usepackage{graphicx}

\begin{document}
\input plot-tmp
\end{document}
```

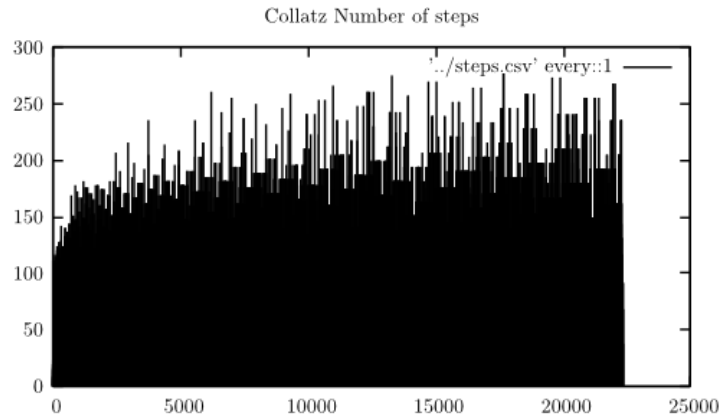
This allows me to plot 22300 data points (but not 22350).

comb plot

To get a comb plot, you have to add "with impulses to the plot command. So your plot.gnuplot file should look like this:

```
set terminal latex
set output "plot-tmp.tex"
set datafile separator ","
set title "Collatz Number of steps"
plot '../steps.csv' every::1 with impulses
```

which produces:



R

You can (should?) plot large data to PDF-files with R. R could handle the 120 MB file with 10,000,000 data points.

To do so, you have to install R and ggplot2 (`sudo apt-get install r-cran-ggplot2`).

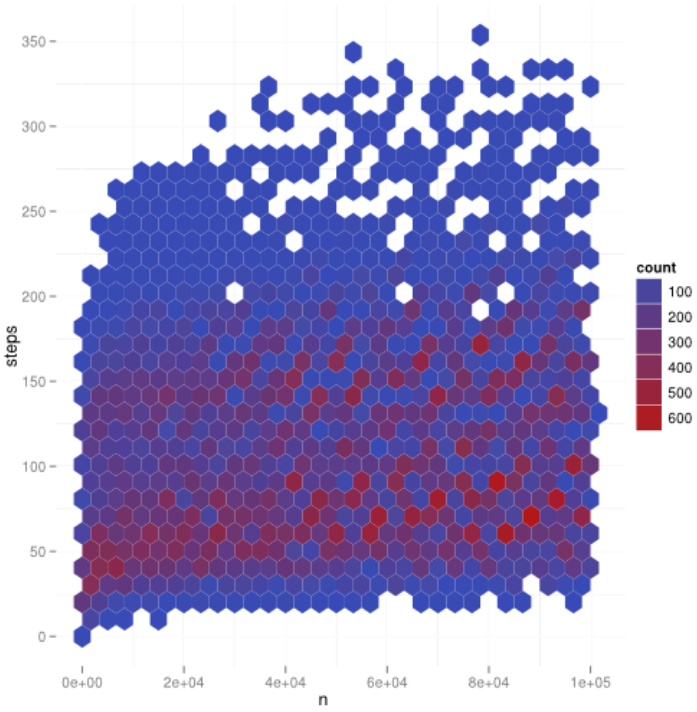
Now you can start R from command line with `sudo R` (it has to be a capital letter) to install hexbin: `install.packages("hexbin", dependencies=TRUE);` (source)

Now create a file called `analyze.R` and copy this:

```
library(ggplot2)

mydata = read.csv("/home/moose/Downloads/algorithms/collatz/steps.csv")

# Prepare data
p<-ggplot(mydata, aes ( x=n,y=steps ))

# Plus means you add those options to the plot
p + geom_hex( bins=30 ) + opts(panel.background = theme_rect(fill='white',
colour='white'))
```

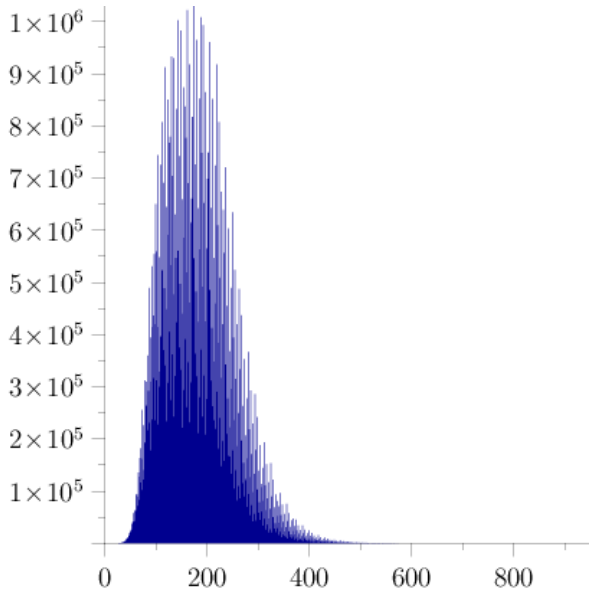
Execute it: `R -f analyze.R`

And you get:



edited May 16 '13 at 12:33

community wiki
5 revs
moose



The formulation of the question looks a bit confusing. First it talks about the graph from `Wikipedia` link, which is an ordinary bar chart that represents a histogram: for every integer `x` (the number of steps to reach `1` from some number from `1..100000000`) there is an integer `y` (the amount of numbers from that interval, that have the same number of steps `x`). Note, that the `x` range is not so big, just less than `600` individual points collect the statistics obtained from `100mln` numbers. Next question is more simple and more difficult at the same time: to plot the number of steps for every number tested in a range. It is simpler, because there is no need to accumulate the amount of numbers with the same `x` steps, while the difficulty is... To plot such a graph with, say, a "humble number" of `100000` points, every point has to be plotted (otherwise it would be just some kind of useless random mess). Assuming that every bar is just one point wide, to print if at `1200dpi`, the total width of the `x`-axis would be roughly **two meters wide**.

So, this `Asymptote` code (without any extra programs) is used to plot the same kind of the graph as that of `Wikipedia` link, a statistics obtained from `nMax=100000000` (100mln) numbers. It took about 20 min to run in the background on a not-very-advanced laptop. The histogram data are accumulated in the integer array `stepCount` and printed to `stdout`. The output also includes:

- `n=100000000` - max number in a range tested,
- `highest=2185143829170100` - max number reached during the whole test,
- `maxStep=949` max number of steps to reach `1`,
- `maxStepArg=63728127` the number, that needs `maxStep` steps to reach `1`.

So, starting from `63728127` it should take `949` steps to get `1`.

`collatz.asy`:

```
import graph;
int nMax=100000000;
int highest=1;
int maxStepArg=1;
int[] arStep=array(nMax,0);

int stepMaxLimit=1000;
int[] stepCount=array(stepMaxLimit,0);

int steps(int nn){
  int n=nn;
  int m=0;
  int nextn;
  while(n>1){
    if((n%2)==0){
      nextn=floor(n/2);
      ++m;
    }else{
      assert(n<floor((intMax-1)/3),"too big n="+string(n));
      nextn=3n+1;
```

```
      highest=max(highest,nextn);
      ++m;
    }
    if(nextn<arStep.length && arStep[nextn-1]>0){
      m+=arStep[nextn-1];
      n=1;
      return m;
    }else{
      n=nextn;
    }
  }
  return m;
}

int maxStep=1;
int j;
for(int i=1;i<=nMax;++i){
  j=steps(i);
  ++stepCount[j];
  if(j>maxStep){
    maxStep=j;
    maxStepArg=i;
  }
  arStep[i-1]=j;
}

write("n="+string(nMax)
+", highest="+string(highest)
+", maxStep = "+string(maxStep)
+", maxStepArg = "+string(maxStepArg)
);

size(256,256,IgnoreAspect);
int nj;
real t;

pen barPen=rgb(0,0,0.5608)+opacity(0.6)+0.8pt;
guide g=(0,0);
for(int i=1;i<=maxStep;++i){
  g=g--(i,stepCount[i])--(i+1,stepCount[i])--(i+1,0);
}
g=g--cycle;

fill(g,barPen);
xaxis(0,maxStep,RightTicks());
yaxis(0,max(stepCount),LeftTicks(beginlabel=false));

write(stepCount[0:maxStep+4]);
```

answered May 18 '13 at 17:00

g.kov
13.7k 1 26 61