

## Sharing your visualizations with others

After you have created your visualizations, you are ready to share them with your collaborators, colleagues, and with others. Here, we will show how you would go about doing final customizations to your figures, and saving them in an appropriate format.

### 2. A figure to share

Take for example this figure that you previously created to display data about the number of gold medals that each of several countries won in the 2016 Olympic Games. When you previously ran this code, it displayed the figure on your screen when you called the `plt-dot-show` method at the end of this code.

### 3. Saving the figure to file

Now, we replace the call to `plt-dot-show` with a call to the Figure object's `savefig` method. We provide a file-name as input to the function. If we do this, the figure will no longer appear on our screen, but instead appear as a file on our file-system called "gold-underscore-medals-dot-png". In the interactive Python shell that we are using here, we can call the `unix ls` function, which gives us a listing of the files in the present working directory. In this case, only the file that we created is present. We can then share this file that now contains the visualization with others.

### 4. Different file formats

In the previous slide, we saved the figure as a PNG file. This file format provides lossless compression of your image. That means that the image will retain high quality, but will also take up relatively large amounts of disk space or bandwidth. You can choose other file formats, depending on your need. For example, if the image is going to be part of a website, you might want to choose the `jpg` format used here, instead. This format uses lossy compression, and can be used to create figures that take up less disk space and less bandwidth. You can control how small the resulting file will be, and the degree of loss of quality, by setting the `quality` key-word argument. This will be a number between 1 and 100, but you should avoid values above 95, because at that point the compression is no longer effective. Choosing the `svg` file-format will produce a vector graphics file where different elements can be edited in detail by advanced graphics software, such as Gimp or Adobe Illustrator. If you need to edit the figure after producing it, this might be a good choice.

### 5. Resolution

Another key-word that you can use to control the quality of the images that you produce is the `dpi` key-word argument. This stands for dots per inch. The higher this number, the more densely the image will be rendered. If you set this number to 300, for example, this will render a fairly high-quality resolution of your image to file. Of course, the higher the resolution that you ask for, the larger the file-size will be.

### 6. Size

Finally, another thing that you might want to control is the size of the figure. To control this, the Figure object also has a function called `set-underscore-size-underscore-inches`. This function takes a sequence of numbers. The first number sets the width of the figure on the page and the second number sets the height of the figure. So setting the size would also determine the aspect ratio of the figure. For example, you can set your figure to be wide and short

### 7. Another aspect ratio

or long and narrow, like here.