ME201/MTH281/ME400/CHE400
Zeros of Bessel Function Derivatives

We write code here to find the n_{th} zero of the derivative of the Bessel function J_m. Here n is a positive integer, and m is any non-negative real number. The basis of the code will be the *Mathematica* routines FindRoot and BesselJZero. BesselJZero[m,n] returns the nth positive zero of J_m. Before writing the code, we look at various special cases to collect ideas about the implementation of the general case. We start by defining a function which will graph both J and J' for any m.

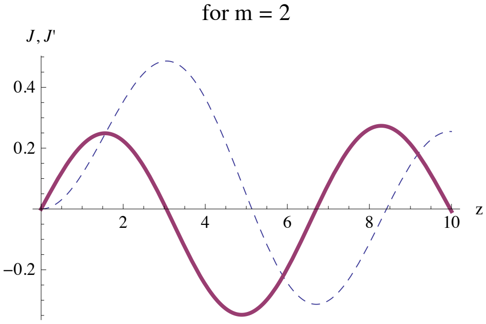
```
In[42]:= bessgraph[m_] := Module[{f, g, z}, f[z_] = BesselJ[m, z]; g[z_] = D[f[z], z];
           Print[Plot[{f[z], g[z]}, {z, 0, 10}, AxesLabel → {"z", "J, J'"},
             PlotStyle → {Dashed, Thick}, PlotLabel → Row[{"Jm(dashed) and J'm(solid)
         for m = ", m}]]]]
```

We set the image size for plots to 250.

```
In[43]:= SetOptions[Plot, ImageSize → 250];
```

We try this for J_2.

```
In[44]:= bessgraph[2]
```



Our first calculation will be to find the second zero of J'_2. From the graph, we see that each zero of the derivative is bracketed by two zeros of the function. We will use those values as initial guesses in FindRoot. More specifically, the second zero of the derivative is bracketed by the first and second positive zeros of the function. Let's get those values which we will call "left" and "right."

```
In[45]:= left = N[BesselJZero[2, 1]]

Out[45]= 5.13562

In[46]:= right = N[BesselJZero[2, 2]]

Out[46]= 8.41724
```

Those values are consistent with what we see from the crossings of the dashed line in the graph. We now use those numbers as initial guesses in a FindRoot search for the second zero of J'_2. Because we are using two initial guesses, FindRoot will use the secant method. (It uses Newton's method for one initial guess.)

```
In[47]:= ans = FindRoot[D[BesselJ[2, z], z] == 0, {z, left, right}]

Out[47]= {z → 6.70613}
```

This looks consistent with the graph. We check our result.

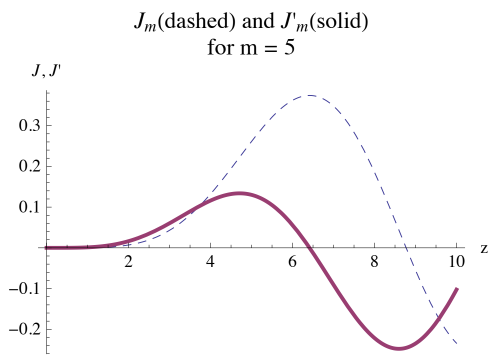
In[48]:= **D[BesselJ[2, z], z] /. ans**

Out[48]= 5.27356×10^{-16}

Close enough to zero.

For most values of m and n the algorithm we have used for this particular case works well. However the special case $n = 1$ (first root) sometimes requires some modifications in the initial guesses. Let's look at a plot for $m = 5$.

In[49]:= **bessgraph[5]**



According to our basic algorithm, the left and right guesses are

In[50]:= **left = 0.0**

Out[50]= 0.

In[51]:= **right = N[BesselJZero[5, 1]]**

Out[51]= 8.77148

In[52]:= **ans = FindRoot[D[BesselJ[5, z], z] == 0, {z, left, right}]**

Out[52]= {z → 0.}

Unfortunately our routine has converged to the unwanted zero at z = 0. The problem is that the function J_5 has a 5th order zero at z = 0. We can see that in the extreme flatness of the curve near the origin. We need to avoid this region in our numerical work, so we choose a left endpoint further to the right. A generic choice which works is

In[53]:= **left = 0.5 * right**

Out[53]= 4.38574

It is clear from the picture that the zero of $J_5{}'$ is between left and right. We try this choice.

In[54]:= **ans = FindRoot[D[BesselJ[5, z], z] == 0, {z, left, right}]**

Out[54]= {z → 6.41562}

Now we have the zero we want. Thus for $n = 1$, we will modify our routine to use these values of left and right.

Other problems occur if m is small. Let's try to find the first zero for $m = 0.1$.

In[55]:= **right = BesselJZero[0.1, 1]**

Out[55]= 2.55745

In[56]:= **left = 0.5 * right**

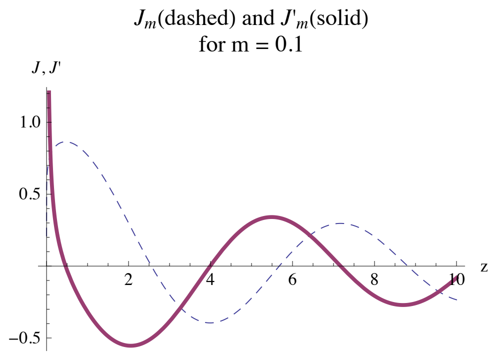
Out[56]= 1.27873

In[57]:= **ans = FindRoot[D[BesselJ[0.1, z], z] == 0, {z, left, right}]**

Out[57]= $\{z \to -7.1731 + 4.00336 \times 10^{-9}\, i\}$

We see two problems here. First we have picked up a numerical artifact -- a small complex part. This is easily fixed by including a Re command in our final code. The more serious problem is that we have found a root that we don't want. Let's graph the function to see what is going on.

In[58]:=
```
bessgraph[0.1]
```



We see that there is a positive zero of the derivative very near the origin. In this case, the zero was not included between left and right, and the root-finding routine wandered off. For small z, we may approximate $J_m(z)$ by the first two terms in the series expansion:

In[59]:=
```
Series[BesselJ[m, z], {z, 0, 2}]
```

Out[59]= $z^m \left(\frac{2^{-m}}{\text{Gamma}[1+m]} - \frac{2^{-2-m} z^2}{(1+m)\,\text{Gamma}[1+m]} + O[z]^3 \right)$

We convert this to a normal expression without the error estimate:

In[60]:=
```
approx = Normal[%]
```

Out[60]= $z^m \left(\frac{2^{-m}}{\text{Gamma}[1+m]} - \frac{2^{-2-m} z^2}{(1+m)\,\text{Gamma}[1+m]} \right)$

We use this approximation to find the location of the first zero of the derivative. First we clean it up a bit. We may multiply or divide by any non-zero constant without changing the location of the zero of the derivative.

In[61]:=
```
approx1 = Simplify[(1 + m) Gamma[1 + m] * 2^(2+m) * approx]
```

Out[61]= $z^m \left(4 + 4\,m - z^2\right)$

In[62]:=
```
Simplify[Solve[D[approx1, z] == 0, z], m > 0]
```

Out[62]= $\left\{\left\{z \to 2 \sqrt{\frac{m\,(1+m)}{2+m}}\right\}\right\}$

We see that for small m, the first zero of the derivative of $J_m$is approximately $\sqrt{2\,m}$. We want to choose our left bracket endpoint to the left of this, so we choose it to be $\sqrt{m}$. Let's see if that works.

In[63]:=
```
right = BesselJZero[0.1, 1]
```

Out[63]= 2.55745

In[64]:=
```
left = Sqrt[0.1]
```

Out[64]= 0.316228

In[65]:=
```
ans = FindRoot[D[BesselJ[0.1, z], z] == 0, {z, left, right}]
```

Out[65]= {z → 0.46351}

Our approximate value was

In[66]:= $2\sqrt{\frac{m\,(1+m)}{2+m}}$ /. m → 0.1

Out[66]= 0.457738

We have almost all of our special cases handled now. The last special case is $m = 0$. We see from the above results that the first positive zero of J'_m goes to zero as m goes to zero. Thus the first zero of the derivative of J_0 should be identified as 0. In this way all of the zeros will depend continuously on m. A second reason for including this value is that in Fourier-Bessel expansions, there is a non-trivial mode associated with it (analogous to the constant term in a Fourier cosine series).

Now we are ready to write our general routine based on FindRoot and BesselJZero. For values of n other than 1, we use the two bracketing zeros of J_m in our FindRoot calculation. For $n = 1$, we implement all of the special cases we have just discussed. We take $m = 0.5$ as the boundary between regular and small values of m. We call our function BesselJPrimeZero[m,n], and it returns the n_{th} zero of the derivative of J_m.

In[67]:=
```
BesselJPrimeZero[m_, n_] := Module[{left, right, z},
  right = N[BesselJZero[m, n]]; Which[(n > 1), (left = N[BesselJZero[m, n - 1]];
    Re[z /. Flatten[FindRoot[D[BesselJ[m, z], z] == 0, {z, left, right}]]]),
   (n == 1), (If[(m == 0), (0.0), (If[(m < 0.5), (left = √m), (left = 0.5 * right)];
      Re[z /. Flatten[FindRoot[D[BesselJ[m, z], z] == 0, {z, left, right}]]])])]]
```

We run some test cases.

In[68]:= `Table[BesselJPrimeZero[0, i], {i, 1, 5}]`

Out[68]= {0., 3.83171, 7.01559, 10.1735, 13.3237}

In[69]:= `Table[BesselJPrimeZero[1, i], {i, 1, 5}]`

Out[69]= {1.84118, 5.33144, 8.53632, 11.706, 14.8636}

In[70]:= `Table[BesselJPrimeZero[2, i], {i, 1, 5}]`

Out[70]= {3.05424, 6.70613, 9.96947, 13.1704, 16.3475}

In[71]:= `Table[BesselJPrimeZero[3, i], {i, 1, 5}]`

Out[71]= {4.20119, 8.01524, 11.3459, 14.5858, 17.7887}

In[72]:= `Table[BesselJPrimeZero[4, i], {i, 1, 5}]`

Out[72]= {5.31755, 9.2824, 12.6819, 15.9641, 19.196}

These all agree with tabulated values (**Handbook of Mathematical Functions,** Abramowitz and Stegun, p. 411, National Bureau of Standards, 3rd printing, 1965). We try this for a few non-integer values of order.

In[73]:= `BesselJPrimeZero[0.5, 1]`

Out[73]= 1.16556

We can check this because $J_{1/2}$ can be expressed in terms of elementary functions.

In[74]:= `BesselJ[1 / 2, x]`

Out[74]= $$\frac{\sqrt{\frac{2}{\pi}}\,\mathrm{Sin}[x]}{\sqrt{x}}$$

In[75]:=
```
D[BesselJ[1 / 2, x], x]
```

Out[75]=
$$\frac{\sqrt{\frac{2}{\pi}}\ \mathrm{Cos}[x]}{\sqrt{x}} - \frac{\mathrm{Sin}[x]}{\sqrt{2\,\pi}\ x^{3/2}}$$

So the zero is the root of tan(x) = 2 x.

In[76]:=
```
FindRoot[Tan[x] == 2 x, {x, 1.16}]
```

Out[76]= $\{x \to 1.16556\}$

It checks. Now we try a smaller order.

In[77]:=
```
BesselJPrimeZero[0.1, 1]
```

Out[77]= 0.46351

In[78]:=
```
D[BesselJ[0.1, z], z] /. z → %
```

Out[78]= 0.

For small order, the first zero of the derivative is approximately equal to the square root of twice the order. We have used this in constructing the left bracket in our root routine. Without this modification, FindRoot wanders off to another root for $n = 1$ and small order. With the bracket chosen in this way, the routine works for very small order.

In[79]:=
```
BesselJPrimeZero[0.01, 1]
```

Out[79]= 0.14195

In[80]:=
```
D[BesselJ[0.01, z], z] /. z → %
```

Out[80]= 0.

In[81]:=
```
BesselJPrimeZero[0.001, 1]
```

Out[81]= 0.0447381

In[82]:=
```
D[BesselJ[0.001, z], z] /. z → %
```

Out[82]= 6.93889×10^{-18}