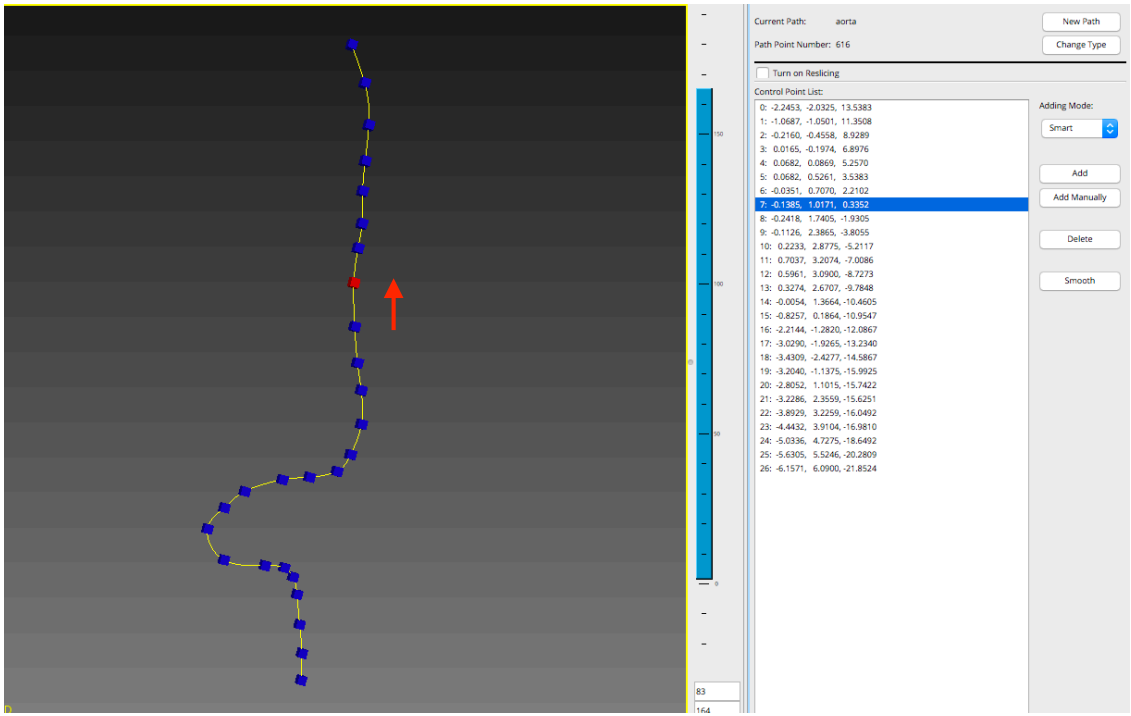
Fitting a cylinder to path control points using the SimVascular Python API

This document describes how to use the SimVascular Python API to fit a cylinder to two control points defined on a path using the **fit_cylinder_to_path.py** Python script.

The **fit_cylinder_to_path.py** script defines a Python class named **PathFitCyl** that encapsulates the data and functions used to create a cylinder aligned with the axis defined by two control points on a path.

A path with control points marked by blue squares. Control points are indexed using integer IDs: 0,1,2,…,26.

Creating a cylinder between controls points 7 (red) and 8 will have an axis aligned with those points (red arrow) and length equal to the distance between those points.



Executing the **fit_cylinder_to_path.py** script defines the **PathFitCyl** class. This class can then be used to create **PathFitCyl** objects. An **PathFitCyl** object is created by given a path name to the object constructor

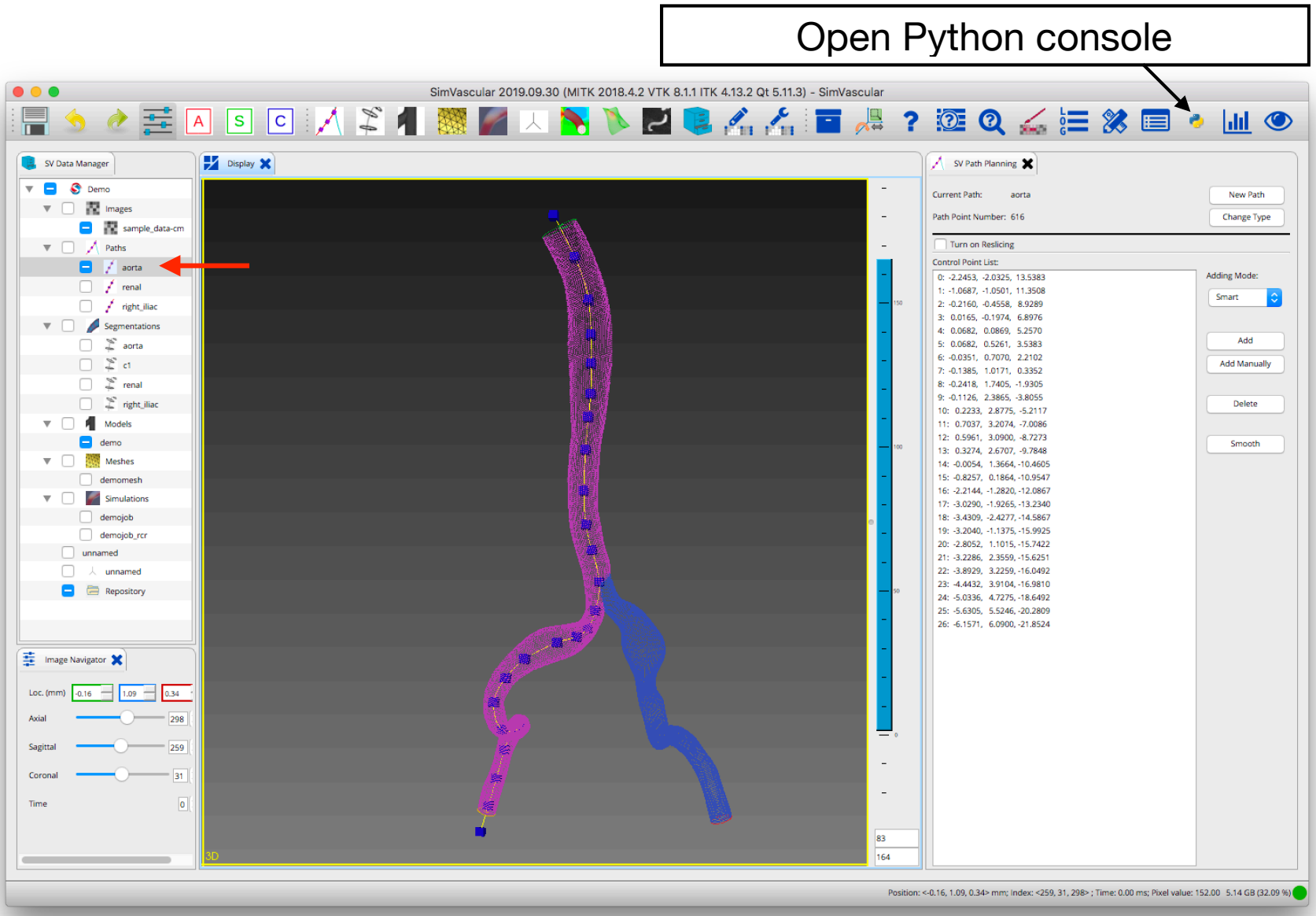
object = PathFitCyl(p*ath_name*), where *path_name* is the name of an SV path.

A cylinder is created using the **fit()** method supplying two control points integer IDs and the cylinder radius

object.fit(id1, id2, radius)

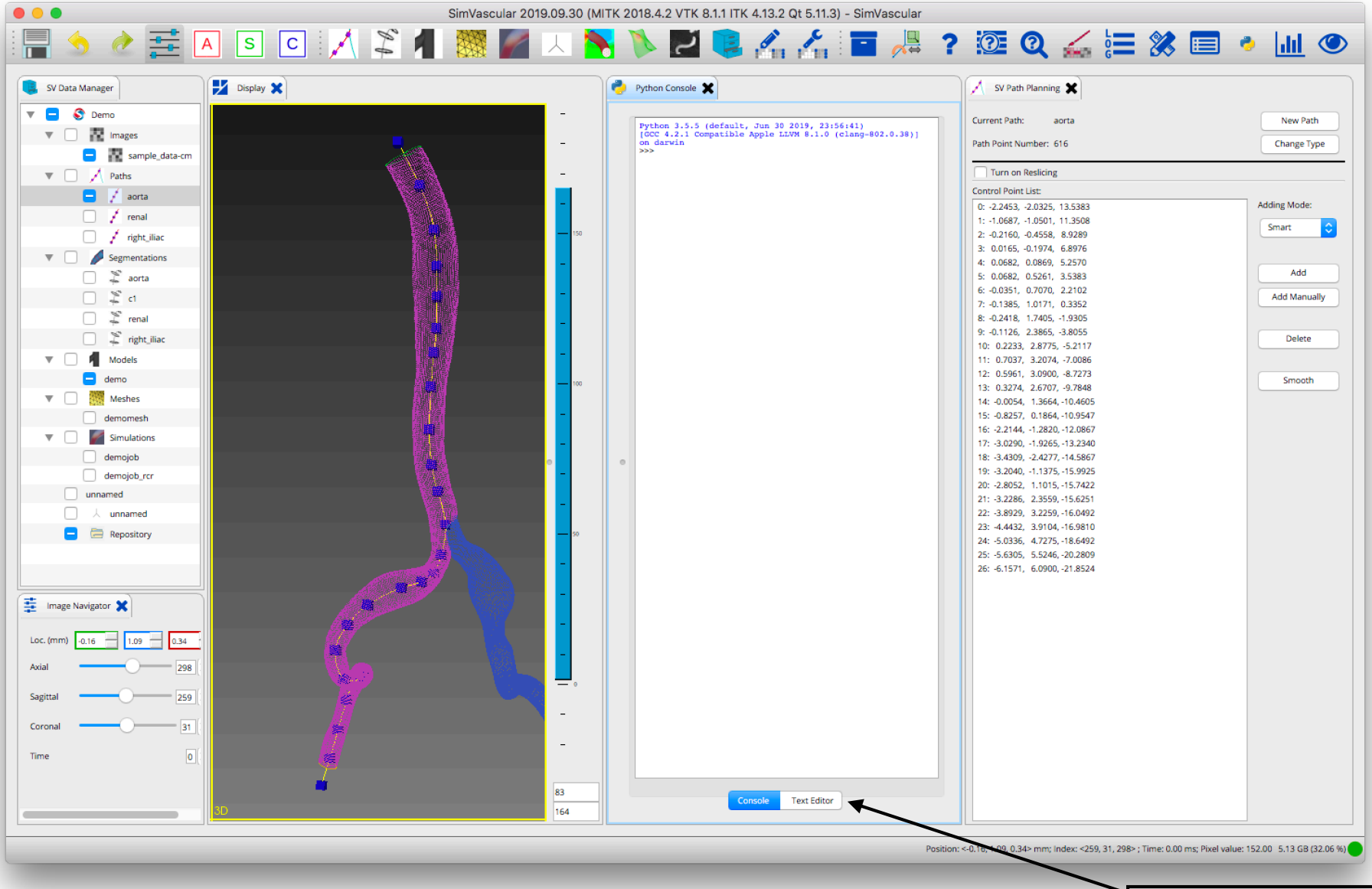
This creates a cylinder and displays it in the SV graphics window.

The following example uses the SimVascular Demo project.



We will use the **aorta** path (red arrow). There are 27 controls points indexed 0,1,2,…26. Control points are shown as blue squares.

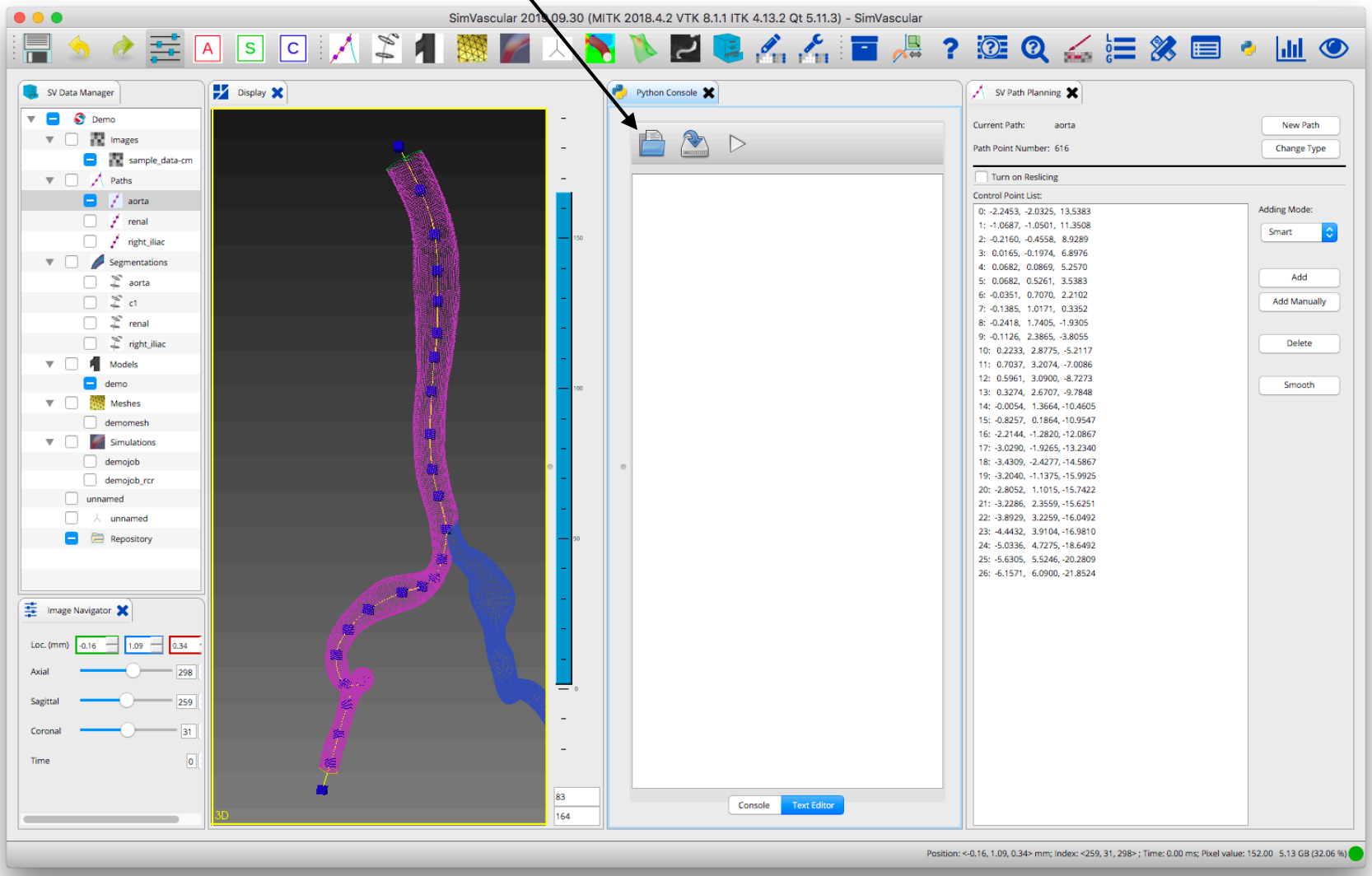
Open the Python console by selecting the Python icon located on the tool bar in the upper right of the SV window.



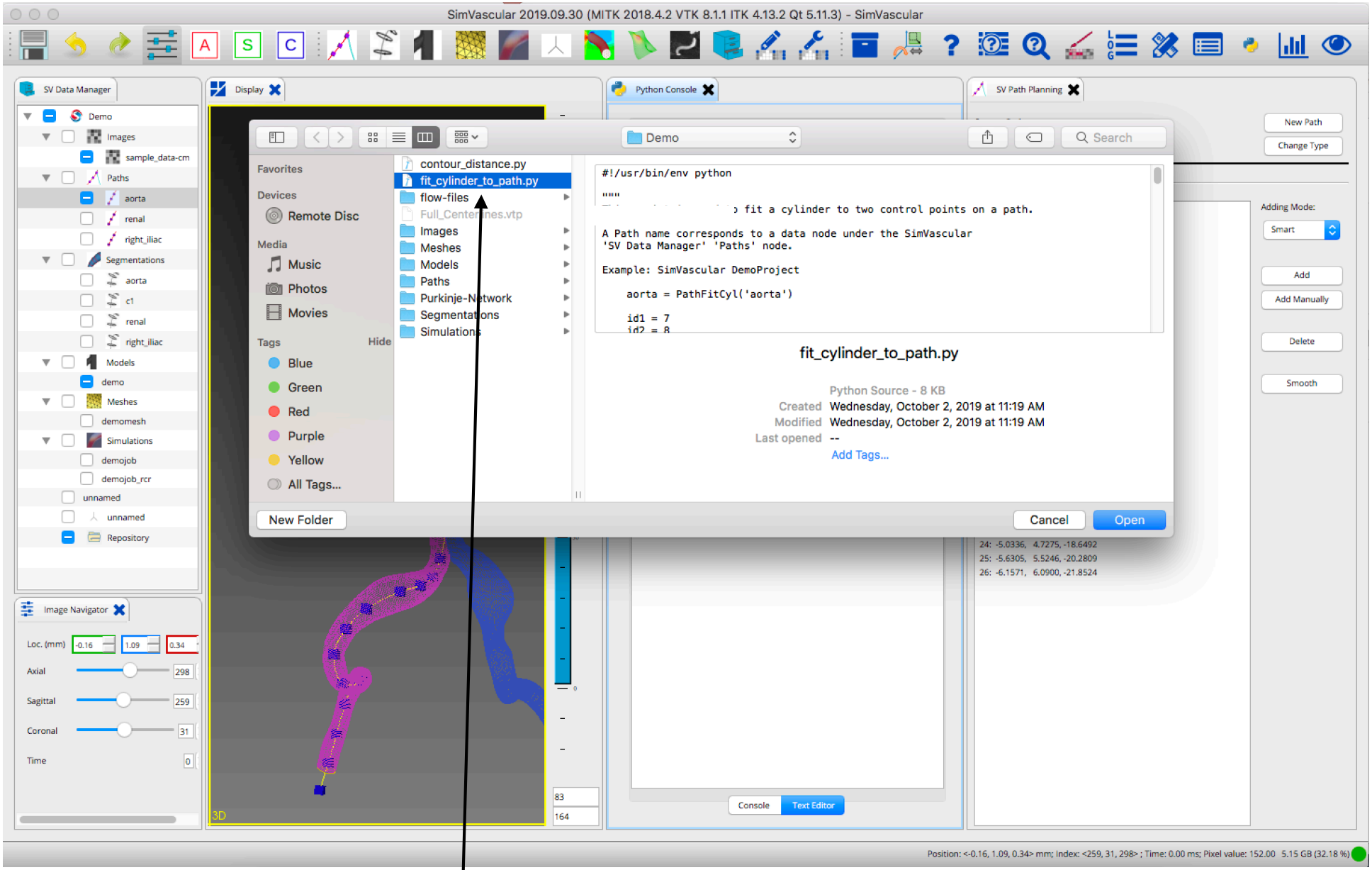
We now want to read in the the **fit_cylinder_to_path.py** Python script.

Select the **Text Editor** button at the bottom of the SV Python Console.

Select to read in a Python script

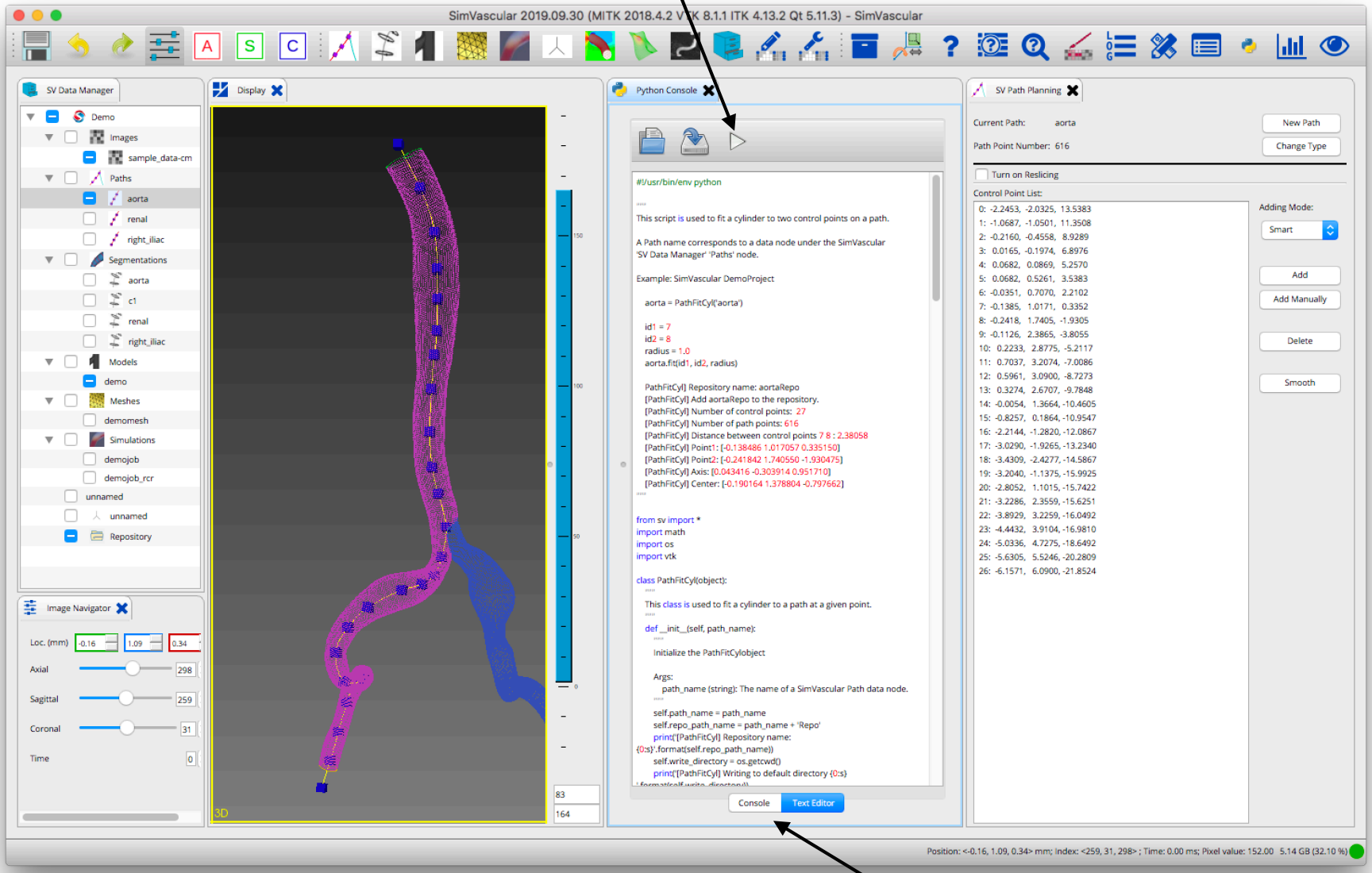


Select the  icon to open a file browser used to select the script.



Select Python the script to read in.

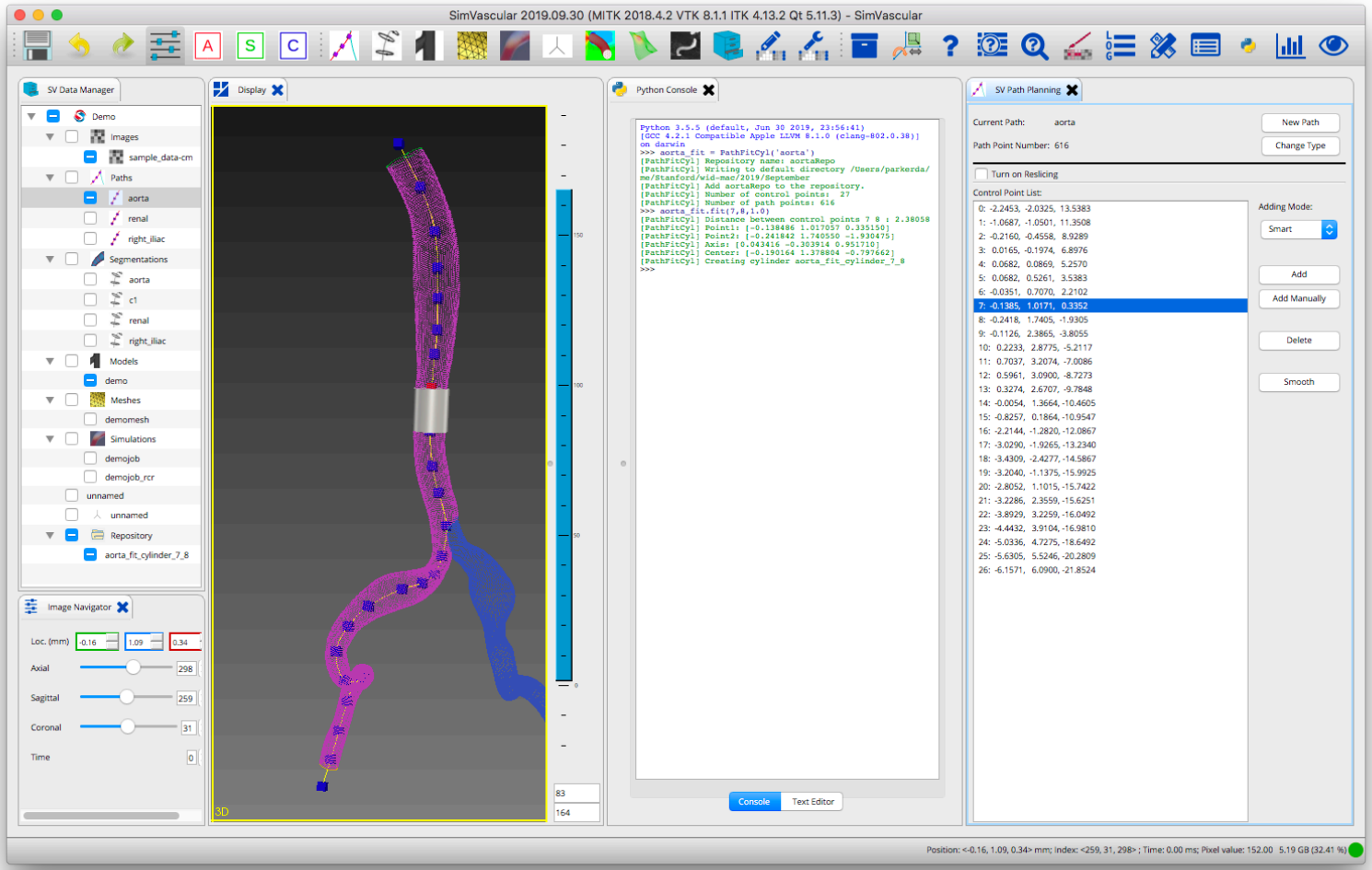
Select the play button to execute the script.



Select the ▷ icon to execute the script which defines the **PathFitCyl** class.

Select **Console** to go back to the Python console.

Select the **Console** button to go back to the Python console.



To create a cylinder type the follow commands into the Python console

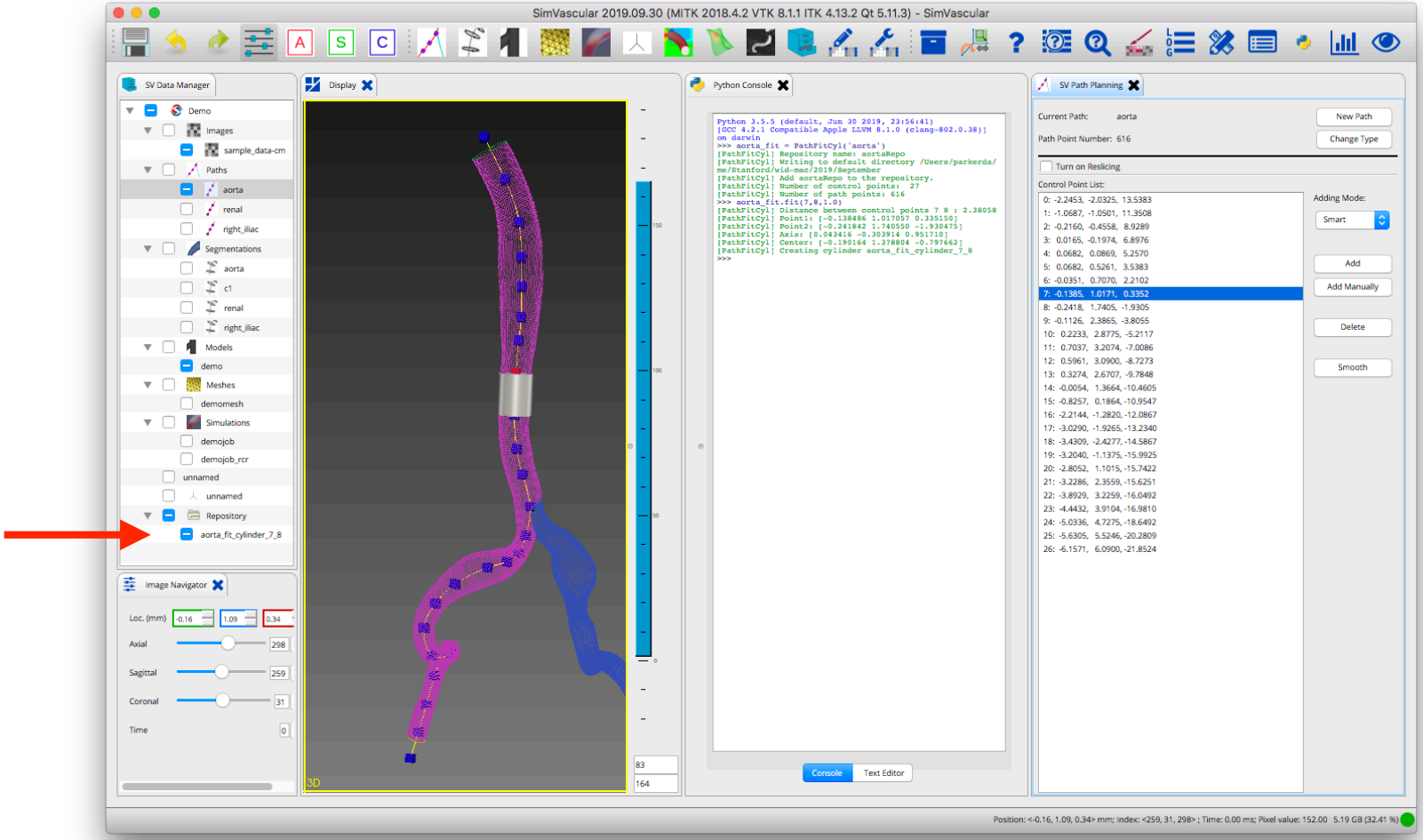
```
>>> aorta_fit = PathFitCyl('aorta')
[PathFitCyl] Repository name: aortaRepo
[PathFitCyl] Writing to default directory /
[PathFitCyl] **** ERROR: Can't write to directory: /
[PathFitCyl]              Set directory using the 'set_write_directory' method.
[PathFitCyl] Add aortaRepo to the repository.
[PathFitCyl] Number of control points:  27
[PathFitCyl] Number of path points: 616
>>> aorta_fit.set_write_directory('/Users/parkerda/')
>>> aorta_fit.fit(7,8,1.0)
[PathFitCyl] Distance between control points 7 8 : 2.38058
[PathFitCyl] Point1: [-0.138486 1.017057 0.335150]
[PathFitCyl] Point2: [-0.241842 1.740550 -1.930475]
[PathFitCyl] Axis: [0.043416 -0.303914 0.951710]
[PathFitCyl] Center: [-0.190164 1.378804 -0.797662]
[PathFitCyl] Creating cylinder aorta_fit_cylinder_7_8
```

Create PathFitCyl object.

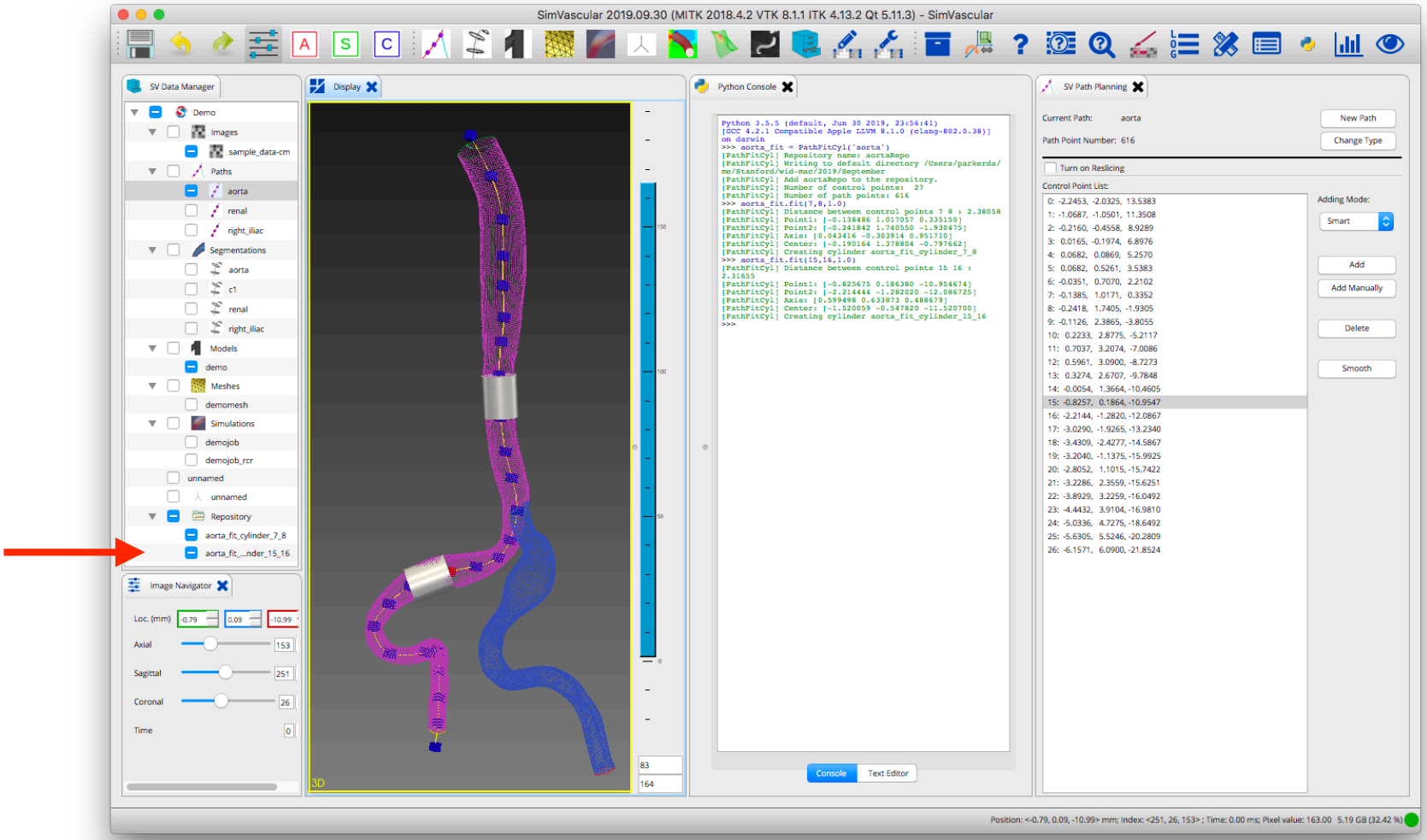
Set the directory used to write cylinder geometry.

The **fit_cylinder_to_path.py** script writes out the cylinder geometry to a VTK .vtp file and then reads it back it in order to display it. This is a work around for a missing API function to create geometry directly.

The directory where the geometry files are written should be set after creating the **PathFitCyl** object using the **set_write_directory()** method. This directory must have write permissions for the current user.

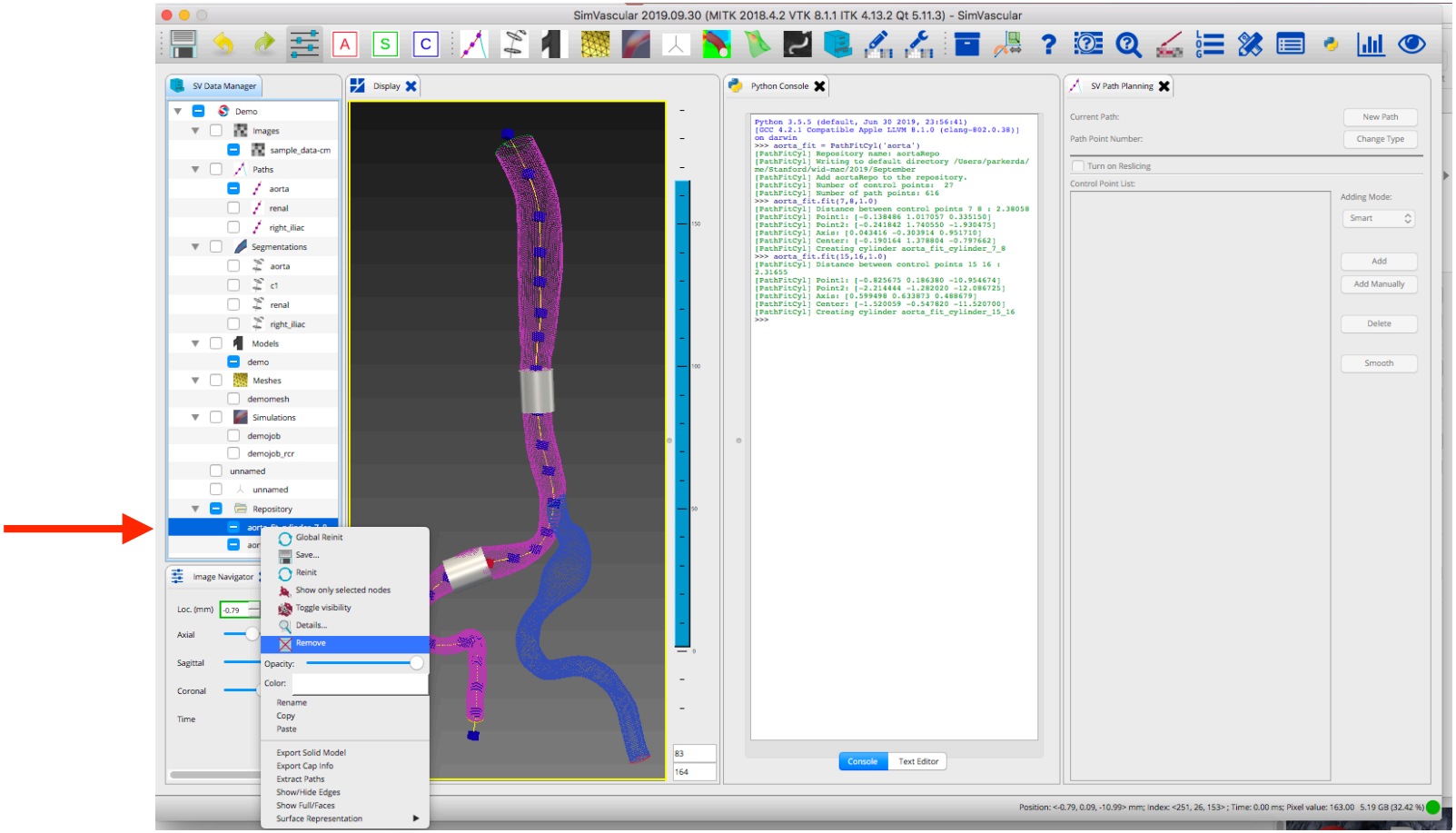


This creates a cylinder named **aorta_fit_cylinder_7_8** under the **Repository** node of **SV Data Manager** (red arrow). The cylinder is displayed in the SV graphics window. Note that the cylinder is not immediately displayed, the graphics window but be zoomed or rotated for the geometry to appear.



Another cylinder can be created using

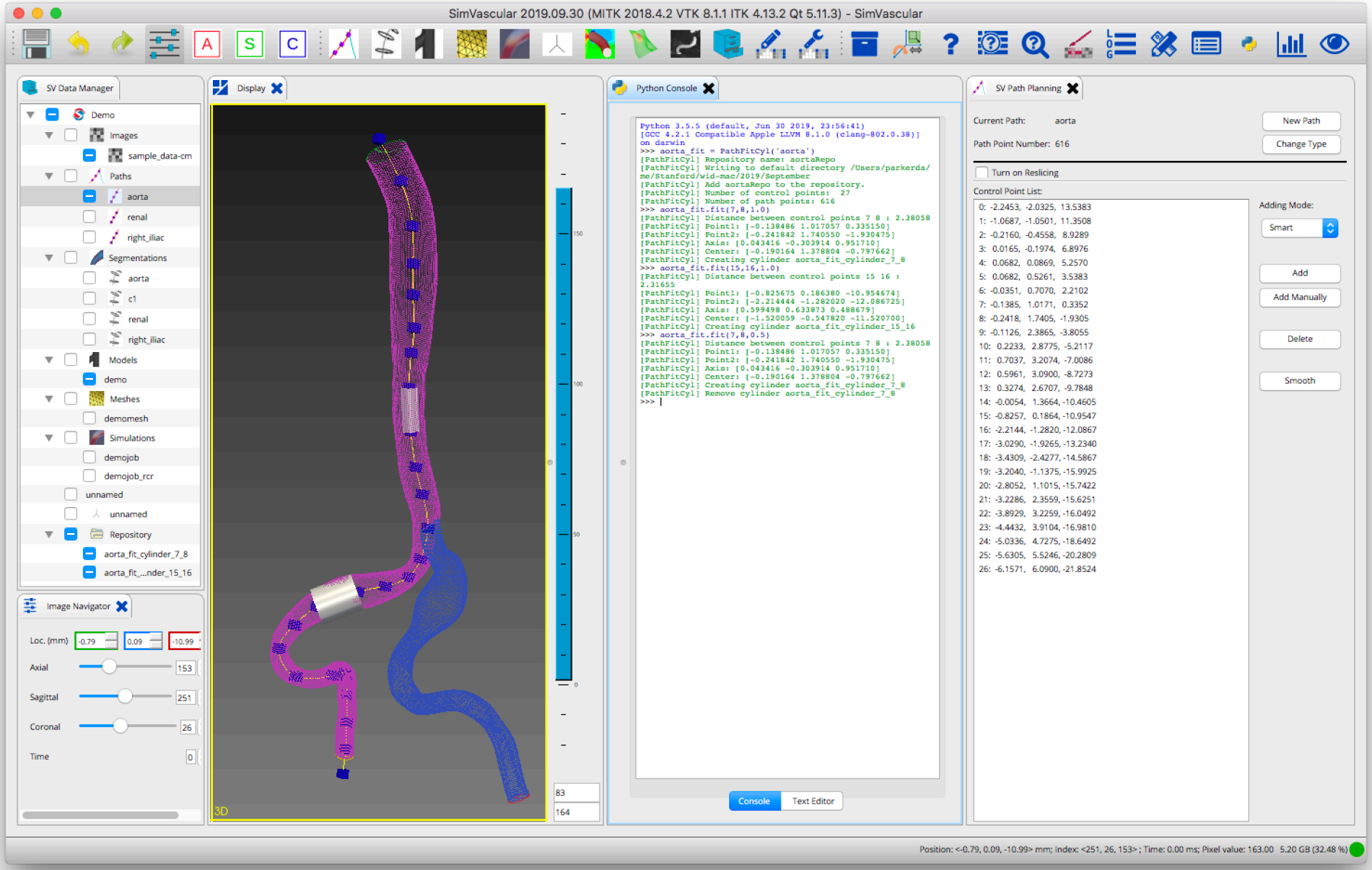
```
>>> aorta_fit.fit(15,16,1.0)
[PathFitCyl] Distance between control points 15 16 : 2.31655
[PathFitCyl] Point1: [-0.825675 0.186380 -10.954674]
[PathFitCyl] Point2: [-2.214444 -1.282020 -12.086725]
[PathFitCyl] Axis: [0.599498 0.633873 0.488679]
[PathFitCyl] Center: [-1.520059 -0.547820 -11.520700]
[PathFitCyl] Creating cylinder aorta_fit_cylinder_15_16
```



To modify the cylinder radius for control points (7,8) delete the **aorta_fit_cylinder_7_8** **Repository** node of **SV Data Manager** (red arrow) by right clicking on **aorta_fit_cylinder_7_8** and selecting the **Remove** menu item.

You can now create another cylinder with a radius of 0.5

```
>>> aorta_fit.fit(7,8,0.5)
[PathFitCyl] Distance between control points 7 8 : 2.38058
[PathFitCyl] Point1: [-0.138486 1.017057 0.335150]
[PathFitCyl] Point2: [-0.241842 1.740550 -1.930475]
[PathFitCyl] Axis: [0.043416 -0.303914 0.951710]
[PathFitCyl] Center: [-0.190164 1.378804 -0.797662]
[PathFitCyl] Creating cylinder aorta_fit_cylinder_7_8
[PathFitCyl] Remove cylinder aorta_fit_cylinder_7_8
```


SimVascular 2019.09.30 (MITK 2018.4.2 VTK 8.1.1 ITK 4.13.2 Qt 5.11.3) - SimVascular
SV Data Manager
Demo
Images
sample_data-cm
Paths
aorta
renal
right_iliac
Segmentations
aorta
c1
renal
right_iliac
Models
demo
Meshes
demomesh
Simulations
demojob
demojob_rcr
unnamed
unnamed
Repository
aorta_fit_cylinder_7_8
aorta_fit_...nder_15_16
Image Navigator
Loc. (mm)
-0.79
0.09
-10.99
Axial
153
Sagittal
251
Coronal
26
Time
0
Display
3D
83
164
Python Console
Python 3.5.5 (default, Jun 30 2019, 23:56:41)
[GCC 4.2.1 Compatible Apple LLVM 8.1.0 (clang-802.0.38)]
on darwin
>>> aorta_fit = PathFitCyl('aorta')
[PathFitCyl] Repository name: aortaRepo
[PathFitCyl] Writing to default directory /Users/parkerda/
me/Stanford/wid-mac/2019/September
[PathFitCyl] Add aortaRepo to the repository.
[PathFitCyl] Number of control points: 27
[PathFitCyl] Number of path points: 616
>>> aorta_fit.fit(7,8,1.0)
[PathFitCyl] Distance between control points 7 8 : 2.38058
[PathFitCyl] Point1: [-0.138486 1.017057 0.335150]
[PathFitCyl] Point2: [-0.241842 1.740550 -1.930475]
[PathFitCyl] Axis: [0.043416 -0.303914 0.951710]
[PathFitCyl] Center: [-0.190164 1.378804 -0.797662]
[PathFitCyl] Creating cylinder aorta_fit_cylinder_7_8
>>> aorta_fit.fit(15,16,1.0)
[PathFitCyl] Distance between control points 15 16 :
2.31655
[PathFitCyl] Point1: [-0.825675 0.186380 -10.954674]
[PathFitCyl] Point2: [-2.214444 -1.282020 -12.086725]
[PathFitCyl] Axis: [0.599498 0.633873 0.488679]
[PathFitCyl] Center: [-1.520059 -0.547820 -11.520700]
[PathFitCyl] Creating cylinder aorta_fit_cylinder_15_16
>>> aorta_fit.fit(7,8,0.5)
[PathFitCyl] Distance between control points 7 8 : 2.38058
[PathFitCyl] Point1: [-0.138486 1.017057 0.335150]
[PathFitCyl] Point2: [-0.241842 1.740550 -1.930475]
[PathFitCyl] Axis: [0.043416 -0.303914 0.951710]
[PathFitCyl] Center: [-0.190164 1.378804 -0.797662]
[PathFitCyl] Creating cylinder aorta_fit_cylinder_7_8
[PathFitCyl] Remove cylinder aorta_fit_cylinder_7_8
>>>
Console
Text Editor
SV Path Planning
Current Path: aorta
Path Point Number: 616
New Path
Change Type
Turn on Reslicing
Control Point List:
0: -2.2453, -2.0325, 13.5383
1: -1.0687, -1.0501, 11.3508
2: -0.2160, -0.4558, 8.9289
3: 0.0165, -0.1974, 6.8976
4: 0.0682, 0.0869, 5.2570
5: 0.0682, 0.5261, 3.5383
6: -0.0351, 0.7070, 2.2102
7: -0.1385, 1.0171, 0.3352
8: -0.2418, 1.7405, -1.9305
9: -0.1126, 2.3865, -3.8055
10: 0.2233, 2.8775, -5.2117
11: 0.7037, 3.2074, -7.0086
12: 0.5961, 3.0900, -8.7273
13: 0.3274, 2.6707, -9.7848
14: -0.0054, 1.3664, -10.4605
15: -0.8257, 0.1864, -10.9547
16: -2.2144, -1.2820, -12.0867
17: -3.0290, -1.9265, -13.2340
18: -3.4309, -2.4277, -14.5867
19: -3.2040, -1.1375, -15.9925
20: -2.8052, 1.1015, -15.7422
21: -3.2286, 2.3559, -15.6251
22: -3.8929, 3.2259, -16.0492
23: -4.4432, 3.9104, -16.9810
24: -5.0336, 4.7275, -18.6492
25: -5.6305, 5.5246, -20.2809
26: -6.1571, 6.0900, -21.8524
Adding Mode:
Smart
Add
Add Manually
Delete
Smooth
Position: <-0.79, 0.09, -10.99> mm; Index: <251, 26, 153> ; Time: 0.00 ms; Pixel value: 163.00 5.20 GB (32.48 %)