Test display structures

November 29, 2016

Code from Simon, 29-11-2016

In [2]:
```
import dicom
import numpy as np
from plotly.graph_objs import *
from plotly.offline import download_plotlyjs, init_notebook_mode, iplot
init_notebook_mode()
```

```
<IPython.core.display.HTML object>
```

In [6]:
```
structures_filename = 'Case7_structures.dcm'
dcm_struct = dicom.read_file(structures_filename, force=True)
```

In [7]:
```
structure_names = [item.ROIName for item in dcm_struct.StructureSetROISeque
structure_names
```

Out[7]:
```
['GTV_P8',
 'GTVn_P2',
 'GTV + 15mm_P1',
 'GTVn + 10mm_P1',
 'CTV65_P12',
 'CTV54_P8',
 'L Parotid_P14',
 'R Parotid_P14',
 'L Lens_P12',
 'R Lens_P12',
 'L Orbit_P12',
 'R Orbit_P12',
 'Brain Stem_P14',
 'Spinal Cord_P14',
 'BS + 3mm_P9',
 'BS + 5mm_P13',
 'Body_P14',
 'Lt Par edit_P14',
 'PTV1_P13',
 'PTV1 edit_P13',
 'PTV2_P13',
```

```
        'PTV2 edit_P13',
        'Pseudo Midline14',
        'Pseudo Oral Ca14',
        'Pseudo PostBrai2',
        'Pseudo Post_P12',
        'Rt Par edit_P14',
        'SC + 3mm_P14',
        'SC + 5mm_P14',
        'Pseudo PTV1_P14',
        'Pseudo PTV2_P14',
        'Left Couch Bar_6',
        'Right Couch Bar6',
        'Artifact_P2',
        'Annulus1_P3',
        'Dose 6175[cGy]_4',
        'Dose 5130[cGy]_4',
        'missing1_P1',
        'missing2_P1']
```

```
In [8]: def pull_structure(number):
            structure_names = [
                item.ROIName for item in dcm_struct.StructureSetROISequence]

            contours_by_slice_raw = [
                item.ContourData for item in dcm_struct.ROIContourSequence[number].
            x = [np.array(item[0::3]) for item in contours_by_slice_raw]
            y = [np.array(item[1::3]) for item in contours_by_slice_raw]
            z = [np.array(item[2::3]) for item in contours_by_slice_raw]

            print("Loaded {}".format(structure_names[number]))
            return x, y, z
```

```
In [9]: def display_structures(list_of_structures, colour_list=None):
            dicom_structure_names = np.array(
                [item.ROIName for item in dcm_struct.StructureSetROISequence])
            combined_trace = []

            for i, structure in enumerate(list_of_structures):
                if colour_list is None:
                    colour = 'black'
                else:
                    colour = colour_list[i]

                reference = (dicom_structure_names == structure)
                if np.all(reference == False):
                    raise Exception("Structure not found (case sensitive)")

                index = int(np.where(reference)[0])
```

```
                x, y, z = pull_structure(index)

                for i in range(len(x)):
                    trace = Scatter3d(
                        x=np.append(x[i], x[i][0]),
                        y=np.append(y[i], y[i][0]),
                        z=np.append(z[i], z[i][0]),
                        mode='lines', line=Line(color=colour, width=3))

                    combined_trace.append(trace)

            iplot(Figure(
                    data=Data(combined_trace),
                    layout=Layout(
                        showlegend=False,
                        width=800, height=1000
                    )
                ))
```

In [10]:
```
display_structures(
             ['Brain Stem_P14', 'Spinal Cord_P14', 'Pseudo PTV1_P14', 'Pseudo PTV2_
             ['Red', 'Blue', 'Green', 'Black']
         )
```

```
Loaded Brain Stem_P14
Loaded Spinal Cord_P14
Loaded Pseudo PTV1_P14
Loaded Pseudo PTV2_P14
```

```
<IPython.core.display.HTML object>
```

In [ ]: