**Computational MR imaging**
**Laboratory 4: Reconstruction of non-Cartesian k-space data**

Report is due on Wednesday before the next lab session at 23:50. Please upload your report on StudOn.

## Preparation

- Install "Pytorch" package on your environment.
- Install "Torch KB-NUFFT toolbox" on your environment.

## Learning objectives

- Reconstruct non-Cartesian MRI data using gridding and NUFFT toolbox
- Apply gridding operations: density compensation, oversampling and deapodization
- Learn to use the NUFFT toolbox

1. **Radial sampling pattern:** Load radial_data.mat (variable) and plot the acquired k-space. Each column corresponds to the readout dimension for each radial line. This data was acquired with a radial acquisition using a golden angle increment (111.246117975°). Generate a sampling trajectory that corresponds to this data for the reconstruction. **Figure 1** shows a plot of the first 10 spokes of such a trajectory for reference. If the matrix size for Cartesian imaging is 384x384, what is the number of radial lines corresponding to the Nyquist rate?

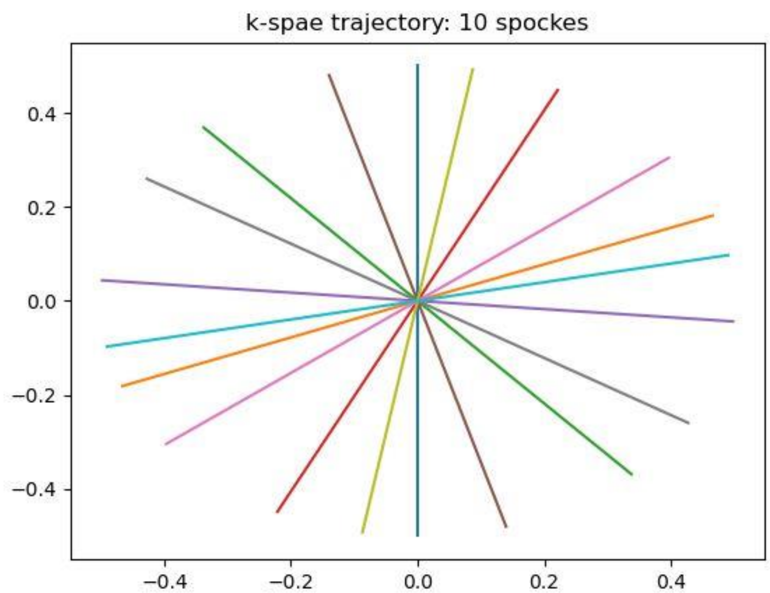


**Figure 1:** Radial trajectory with golden angle increment. Note that the first angle is at pi/2.

2. **Basic gridding reconstruction:** Reconstruct this dataset using the provided grid1 function that grids 2D non-Cartesian k-space data to Cartesian k-space data using triangular gridding kernel of width 2:

```
def grid(d, k, n):
  """Grid non-cartesian kspace data to a cartesion grid
  Keyword Arguments:
    d - 2D numpy array, non-cartesian kspace
    k - 2D numpy array, kspace trajectory, scaled -0.5 to 0.5
    n - int, grid size
  Returns:
    2D numpy array (n, n)
```

Use a 384x384 Cartesian grid. Comment on the artifacts. Can you guess what organ was imaged?

3. **Density compensation**: Reconstruct the radial dataset from part 2 using a ramp filter. Plot your results. Do you need to employ oversampling and de-apodization on this dataset? Explain your answer.

4. **Oversampling**: Grid the decimated data from part 3 using oversampling factors of 1.5 and 2, apply inverse FFT and crop in the image domain.

5. **De-apodization**: Compute the de-apodization function in the image domain and apply to the gridded image with oversampling of 2. (Hint 1: think of the convolution of the delta function.)

$$\hat{m}(x,y) = \frac{\mathbf{1}}{\boldsymbol{c(x,y)+a}}\left\{\left[\left(m(x,y) * s(x,y)\right)c(x,y)\right] * \mathrm{III}\left(\frac{x}{FOV_x}, \frac{y}{FOV_y}\right)\right\}$$

Apodization

Deapodize

6. **NUFFT toolbox**: Reconstruct the radial dataset using a widely used NUFFT toolbox from the research community. Plot your reconstructions and compare them with gridding reconstruction using the triangular kernel.
   a. You will use the Torch KB-NUFFT toolbox from Matt Muckley (Facebook AI research). This toolbox is modeled after Jeffrey Fessler's toolbox. You can get it from github, and documentation in provided there: https://github.com/mmuckley/torchkbnufft

Signal after deapodization is recovered and hence, edges are brighter while center seems a little darker. Check streaking artifacts at edges to see the effects