import matplotlib as mpl

import matplotlib.colors as colors

x\_axis =y\_axis= np.arange(30,100,step=0.5)

# print(len(x\_axis))

# print(theta\_0,theta\_1,theta\_2)

X,Y = np.meshgrid(x\_axis,y\_axis)

z = theta\_0 + theta\_1\*X + theta\_2 \*Y

# temp=[]

# for i in range(len(x\_axis)):

#   for j in range(len(y\_axis)):

#     temp.append(theta\_0 + theta\_1\*x\_axis[i] + theta\_2 \*y\_axis[j])

# z=np.array(temp)

# print(z)

# print(len(z))

sigmoid = 1/(1+np.exp(-z))

# print(sigmoid)

# print(len(sigmoid))

# def truncate\_colormap(cmap, minval=0.0, maxval=1.0, n=100):

#     new\_cmap = colors.LinearSegmentedColormap.from\_list(

#         'trunc({n},{a:.2f},{b:.2f})'.format(n=cmap.name, a=minval, b=maxval),

#         cmap(np.linspace(minval, maxval, n)))

#     return new\_cmap

# arr = np.linspace(30,100,num=10000).reshape((100, 100))

fig, ax = plt.subplots(1)

# new\_cmap1 = truncate\_colormap(plt.get\_cmap('jet'), 0.45, 1.0)

# new\_cmap2 = truncate\_colormap(plt.get\_cmap('brg'), 1.0, 0.45)

# ax.imshow(sigmoid,cmap="bwr\_r")

# ax.imshow(arr,origin='lower',cmap="bwr\_r")

ax.imshow(sigmoid,origin='lower',cmap="bwr\_r")

# ax.imshow(sigmoid,origin='lower',cmap="seismic\_r",interpolation='nearest')

# ax[0].imshow(sigmoid,aspect='auto', cmap = new\_cmap1)

# ax[1].imshow(a,aspect='auto', cmap = new\_cmap2)

# my\_cmap2 = colors.LinearSegmentedColormap('dd',cdict2,256)

# def colorFader(c1,c2,mix): #fade (linear interpolate) from color c1 (at mix=0) to c2 (mix=1)

#     c1=np.array(mpl.colors.to\_rgb(c1))

#     c2=np.array(mpl.colors.to\_rgb(c2))

#     #print(mpl.colors.to\_hex((1-mix)\*c1 + mix\*c2))

#     return mpl.colors.to\_hex((1-mix)\*c1 + mix\*c2)

# c1='blue'

# c2='red'

# fig, ax = plt.subplots(figsize=(8, 5))

# for x in sigmoid:

#   ax.axvline(x\_axis, color=colorFader(c1,c2,x), linewidth=4)

#     # ax.axvline(x, color=colorFader(c1,c2,x/n), linewidth=4)

# plt.figure()

# a=np.outer(np.arange(0,1,0.01),np.ones(10))

# cdict2 = {'red':   [(0.0,  0.0, 0.0),

#                    (0.5,  1.0, 1.0),

#                    (1.0,  1.0, 1.0)],

#          'green': [(0.0,  0.0, 0.0),

#                    (0.25, 0.0, 0.0),

#                    (0.75, 1.0, 1.0),

#                    (1.0,  1.0, 1.0)],

#          'blue':  [(0.0,  0.0, 0.0),

#                    (0.5,  0.0, 0.0),

#                    (1.0,  1.0, 1.0)]}

# my\_cmap2 = colors.LinearSegmentedColormap('my\_colormap2',cdict2,256)

# plt.imshow(sigmoid,aspect='auto', cmap =my\_cmap2)

# plt.show()

plt.xlabel('x')

plt.ylabel('y')

# plt.scatter(x\_axis,y\_axis,sigmoid)

plt.scatter(x\_label0, y\_label0,label="l=0", c='b')

plt.scatter(x\_label1, y\_label1,label="l=1", c='r')

# plt.plot(x\_axis,y\_axis)

plt.legend()

plt.show()