

ひとり情シスのIです。

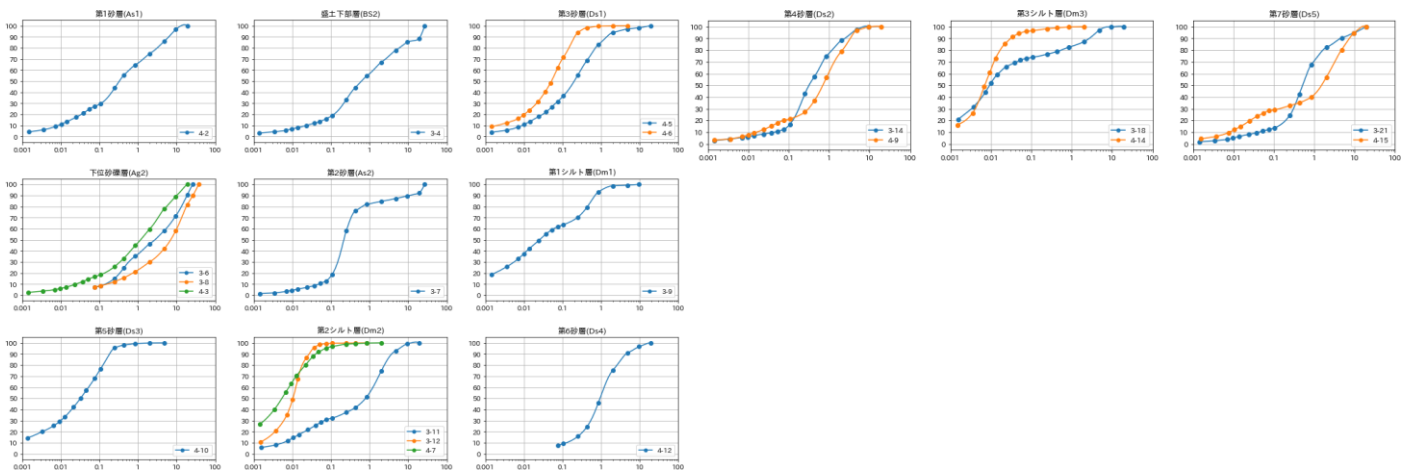
今日は、粒径加積のグラフを、縦横何個ずつ表示させるかを与えたら、グラフの数に応じて自動的にビットマップを分割して生成する処理と、ページサイズに応じてビットマップの大きさを変える等の処理を埋め込んでみました。あとは、GUIを作って、各種設定をそこから設定できればと思います。

```

114 #DBのテーブルに書き込まれた粒径加積のデータから粒径加積曲線を作成する。
115 def ryukeikasekiDraw():
116
117     try:
118         #DBアクセス
119         con = sqlite3.connect("geodata.db3",isolation_level='EXCLUSIVE')
120         cur=con.execute("select distinct 土質名,土質記号 from 試料採取_粒径加積 order by 開始深度")
121         soudata = cur.fetchall()
122
123         A4_tate_x=(150/25.4)*1.25
124         A4_tate_y=(247/25.4)*1.25
125         A4_yoko_x=(247/25.4)*1.25
126         A4_yoko_y=(150/25.4)*1.25
127         A3_yoko_x=(360/25.4)*1.25
128         A3_yoko_y=(247/25.4)*1.25
129         tate=3
130         yoko=3
131         graph_per_page=tate*yoko
132
133         #複数ページにまたがった際にデータを1ページごとに分割する。
134         split_soudata = [soudata[idx:idx + graph_per_page] for idx in range(0,len(soudata), graph_per_page)]
135
136         for i,tmp_soudata in enumerate(split_soudata):
137             fig = plt.figure(figsize=(A3_yoko_x,A3_yoko_y))
138             for sou_no,soutap in enumerate(tmp_soudata):
139                 #サンプル番号を取得する。
140                 sou = list(soutap)
141                 sql = "select 連番, 試料番号 from 試料採取_粒径加積 where 土質名 = ? and 土質記号 = ? order by 試料番号"
142                 cur=con.execute(sql,(sou[0],sou[1]))
143                 sou2=cur.fetchall()
144
145                 regend_lines=[]
146                 for j,soutap in enumerate(sou2):
147                     sou3 = list(soutap)
148                     sql2="select 粒径,パーセント from 曲線データ_粒径加積 where 試料連番="+str(sou3[0])
149                     cur=con.execute(sql2)
150                     sousou=cur.fetchall()
151                     arr = np.asarray(sousou)
152                     ar=arr.T
153                     d1 = ar[0:2,: ]
154                     d1=d1[:,np.all(d1!=None, axis=0)]
155                     x1=sorted(d1[0])
156                     y1=sorted(d1[1])
157
158                     regend_lines=partgraph(x1,y1,tate,yoko,sou_no+1,sou[0]+'('+sou[1]+')',sou3[1],fig,j,len(sou2),regend_lines)
159
160                 #出力は200dpiで(画像の縦横のサイズは変わらない)
161                 fig.savefig("ryukei_"+str(i+1)+".png",bbox_inches='tight', pad_inches=0,dpi=200)
162
163     except Exception as e:

```

下は、12個のグラフを、1ページあたり縦3つ、横3つでA3サイズのビットマップで分割して生成したものです(左が1枚目、右が2枚目)。9個ごとにリスト分割をして、順に渡しています。



とりあえず、今日はこんなところで。