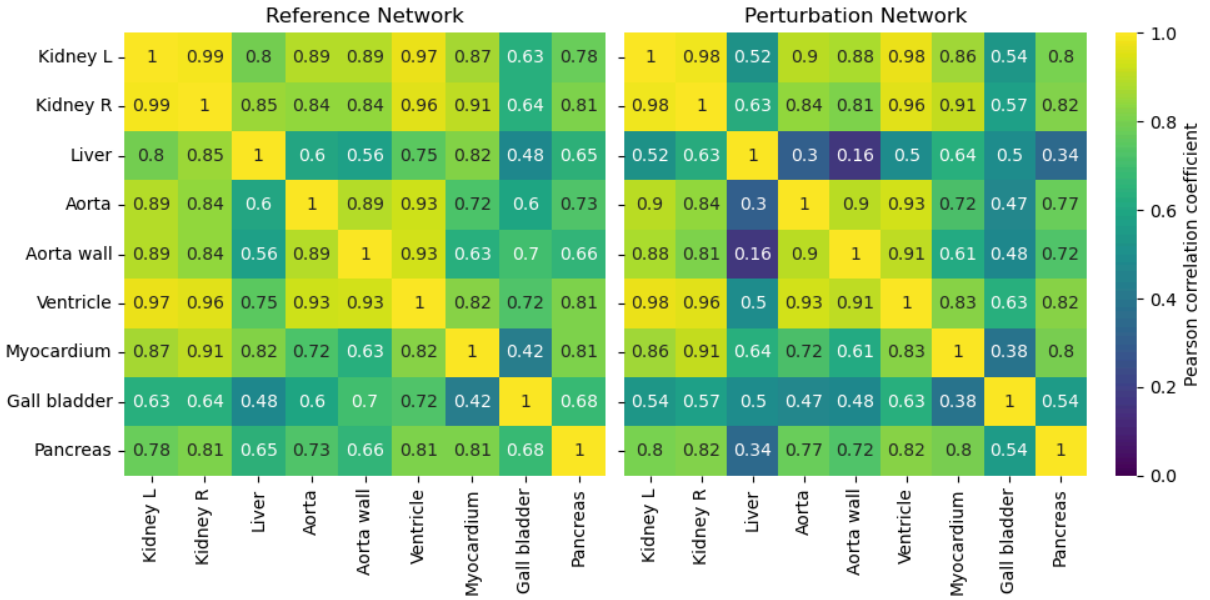
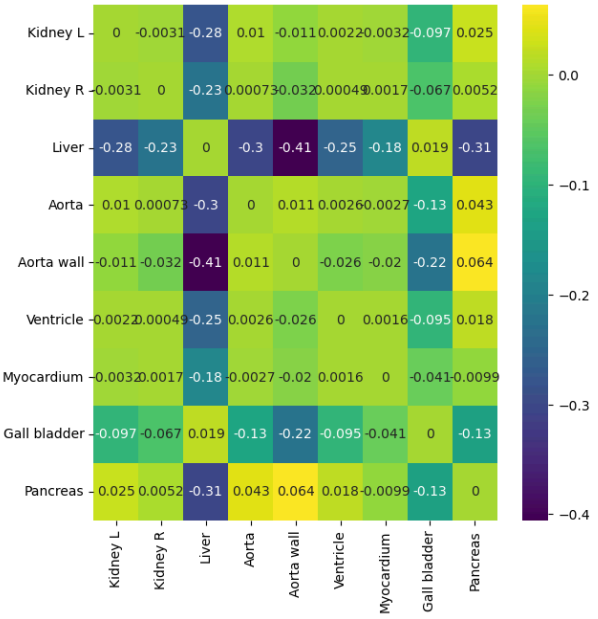
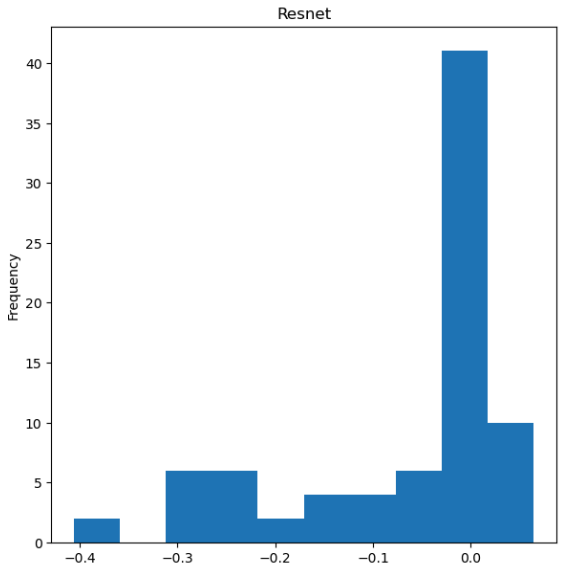
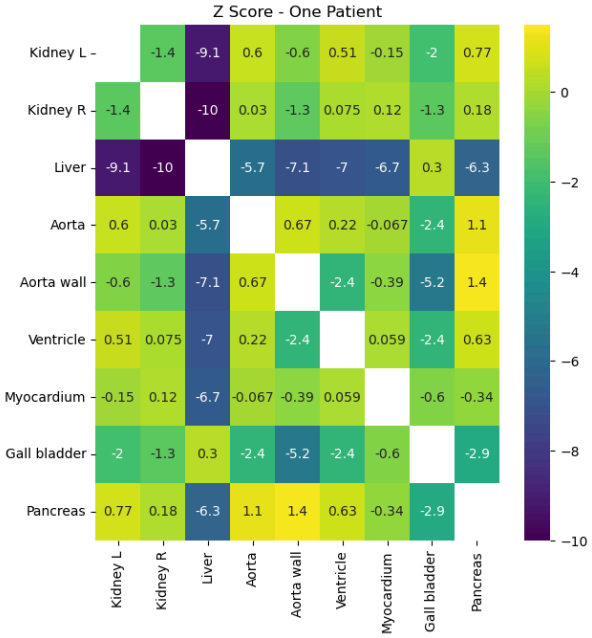
In [3]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import networkx as nx
from netgraph import Graph

#color palette choice for all graphing
palette='viridis'

#read in file, create dataframe with subjects on y, regions on x
filename = 'IsotoPK_data_linearinterpolated_static.csv'
df = pd.read_csv(filename, sep=',', usecols=['Subject', 'Kidney L', 'Kidney
                                             'Gall bladder', 'Pancreas'], index_col
#df = df.transpose() swaps to subjects on x, regions on y

#get one dataframe of control subjects only
ctrlsubs = ['S00732', 'S00780', 'S00786', 'S00793', 'S00796', 'S00801', 'S00
                    'S00956']
rifsubs = ['S00712', 'S00724', 'S00733', 'S00781', 'S00794', 'S00802', 'S008
ctrl = df.transpose()[ctrlsubs].copy().transpose()

#create the refNET with PCC between each region pair, subjects are the x-axi
#paper uses partial PCC with covariates of age and gender, i will just use P
#(Identifying the individual metabolic abnormities from a systemic perspecti
refnet = ctrl.corr(method='pearson')
refnet = refnet.transpose()

#create the plot with two subplots, setting figure size and gridspec ratio (
fig, ax = plt.subplots(ncols=2, figsize=(10,5),sharey=True, gridspec_kw={'wi

#create the heatmap to represent the correlation network of the control subj
snsctrl = sns.heatmap(refnet, annot=True, vmax=1.0, vmin=0.0, cbar_kws={'lab
                      cmap=palette)
ax[0].set_title('Reference Network')

#create the perturbation network by adding one rif subject to the ctrl cohor
ptb = df.transpose()[ctrlsubs].copy()
ptb = ptb.assign(rifsub=df.transpose()[rifsubs[0]])
ptbnet = ptb.transpose().corr(method='pearson').transpose()

#create the heatmap that represents the ptbNET
snsptb = sns.heatmap(ptbnet, annot=True, vmax=1.0, vmin=0.0, cbar_kws={'labe
                      cmap=palette)
ax[1].set_title('Perturbation Network')

#create the resNET=ptbNET-refNET
resnet = ptbnet-refnet
#set threshold of 30% to remove weak residual correlations arising after sub
# thresh = 0.3*max(resnet.stack().where(resnet.stack()>0).max(), abs(resnet.
# print(thresh)
# for col in resnet.columns:
#     resnet.loc[((resnet[col]>0)&(resnet[col]<thresh)) | ((resnet[col]<0)&(

#plot with minimal white space
```

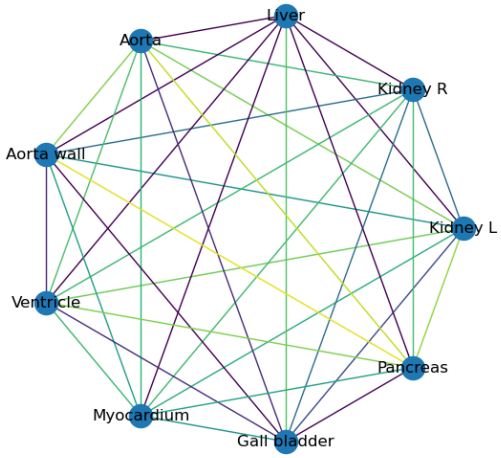
```
fig.tight_layout()
plt.show()

fig, ax = plt.subplots(ncols=2, figsize=(15,7))

snsres = sns.heatmap(resnet, annot=True, cmap=palette, ax=ax[0])

resnet.stack().plot.hist(ax=ax[1])
plt.title('Resnet')
plt.show()

#create the zscore matrix - requires large n??
# zscore = resnet/((1-refnet**2)/(len(ctrl.columns)-1))
# #print(zscore)
# snszscore = sns.heatmap(zscore, annot=True)

#create the zscore plot with two subplots, setting figure size
fig, ax = plt.subplots(ncols=2, figsize=(15,7))

#create zscore matrix without relying on large n
#zscore = (resnet-resnet.stack().mean())/resnet.stack().std()
#zscore = (resnet-resnet.mean())/resnet.std()
zscore = resnet/((1-refnet**2)/(len(ctrlsubs)-1))
snszscore = sns.heatmap(zscore, annot=True, vmin=-10, vmax=1.5, cmap=palette
ax[0].set_title('Z Score - One Patient')

#save to csv
zscore.to_csv('z_score.csv')

#networkx
G = nx.from_pandas_adjacency(zscore, create_using=nx.Graph)
G.remove_edges_from(nx.selfloop_edges(G))
edges, weights = zip(*nx.get_edge_attributes(G, 'weight').items())

max=zscore.stack().max()
nx.draw_circular(G, edge_color=weights, edge_cmap=plt.get_cmap(palette), edg
#plt.colorbar(plt.cm.ScalarMappable(cmap=plt.get_cmap(palette), norm=plt.Nor

plt.show()
```


Reference Network
Perturbation Network
Kidney L 1 0.99 0.8 0.89 0.89 0.97 0.87 0.63 0.78
Kidney R 0.99 1 0.85 0.84 0.84 0.96 0.91 0.64 0.81
Liver 0.8 0.85 1 0.6 0.56 0.75 0.82 0.48 0.65
Aorta 0.89 0.84 0.6 1 0.89 0.93 0.72 0.6 0.73
Aorta wall 0.89 0.84 0.56 0.89 1 0.93 0.63 0.7 0.66
Ventricle 0.97 0.96 0.75 0.93 0.93 1 0.82 0.72 0.81
Myocardium 0.87 0.91 0.82 0.72 0.63 0.82 1 0.42 0.81
Gall bladder 0.63 0.64 0.48 0.6 0.7 0.72 0.42 1 0.68
Pancreas 0.78 0.81 0.65 0.73 0.66 0.81 0.81 0.68 1
1 0.98 0.52 0.9 0.88 0.98 0.86 0.54 0.8
0.98 1 0.63 0.84 0.81 0.96 0.91 0.57 0.82
0.52 0.63 1 0.3 0.16 0.5 0.64 0.5 0.34
0.9 0.84 0.3 1 0.9 0.93 0.72 0.47 0.77
0.88 0.81 0.16 0.9 1 0.91 0.61 0.48 0.72
0.98 0.96 0.5 0.93 0.91 1 0.83 0.63 0.82
0.86 0.91 0.64 0.72 0.61 0.83 1 0.38 0.8
0.54 0.57 0.5 0.47 0.48 0.63 0.38 1 0.54
0.8 0.82 0.34 0.77 0.72 0.82 0.8 0.54 1
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
Pearson correlation coefficient
1.0
0.8
0.6
0.4
0.2
0.0


Kidney L 0 -0.0031 -0.28 0.01 -0.011 0.0022 0.0032 -0.097 0.025
Kidney R -0.0031 0 -0.23 0.00073 -0.032 0.00049 0.0017 -0.067 0.0052
Liver -0.28 -0.23 0 -0.3 -0.41 -0.25 -0.18 0.019 -0.31
Aorta 0.01 0.00073 -0.3 0 0.011 0.0026 0.0027 -0.13 0.043
Aorta wall -0.011 -0.032 -0.41 0.011 0 -0.026 -0.02 -0.22 0.064
Ventricle 0.0022 0.00049 -0.25 0.0026 -0.026 0 0.0016 -0.095 0.018
Myocardium 0.0032 0.0017 -0.18 0.0027 -0.02 0.0016 0 -0.041 -0.0099
Gall bladder -0.097 -0.067 0.019 -0.13 -0.22 -0.095 -0.041 0 -0.13
Pancreas 0.025 0.0052 -0.31 0.043 0.064 0.018 0.0099 -0.13 0
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
0.0
−0.1
−0.2
−0.3
−0.4

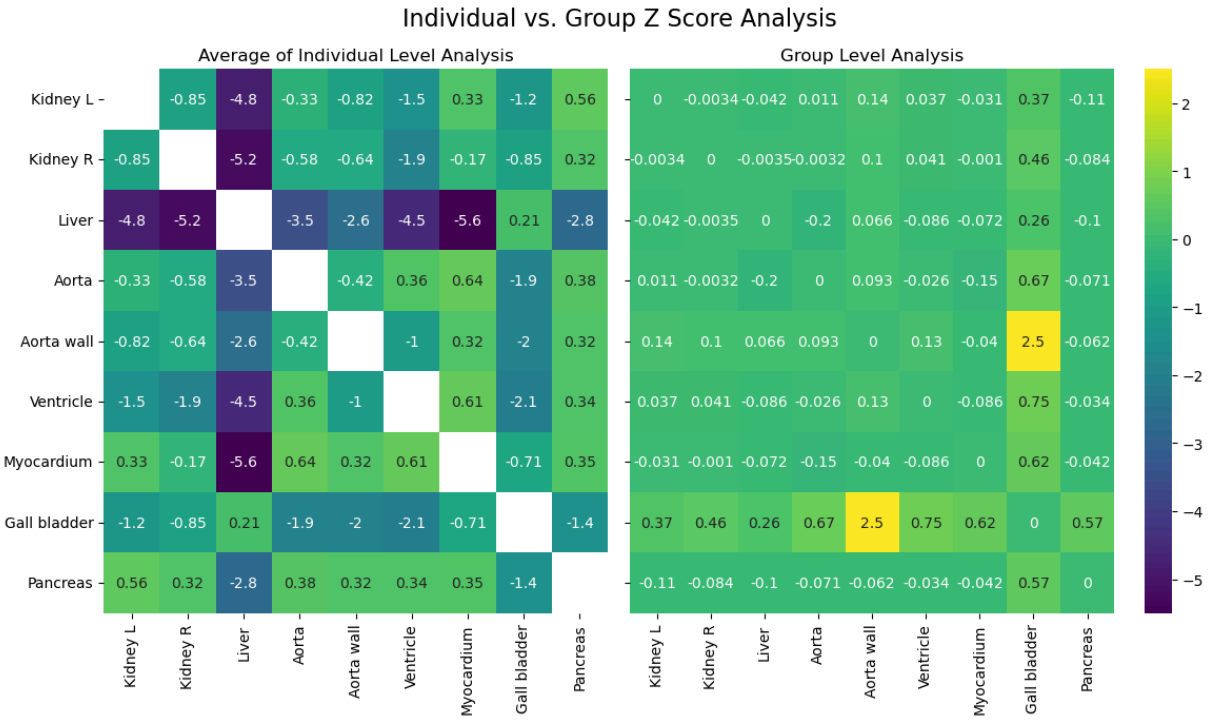

Resnet
Frequency
40
35
30
25
20
15
10
5
0
−0.4
−0.3
−0.2
−0.1
0.0



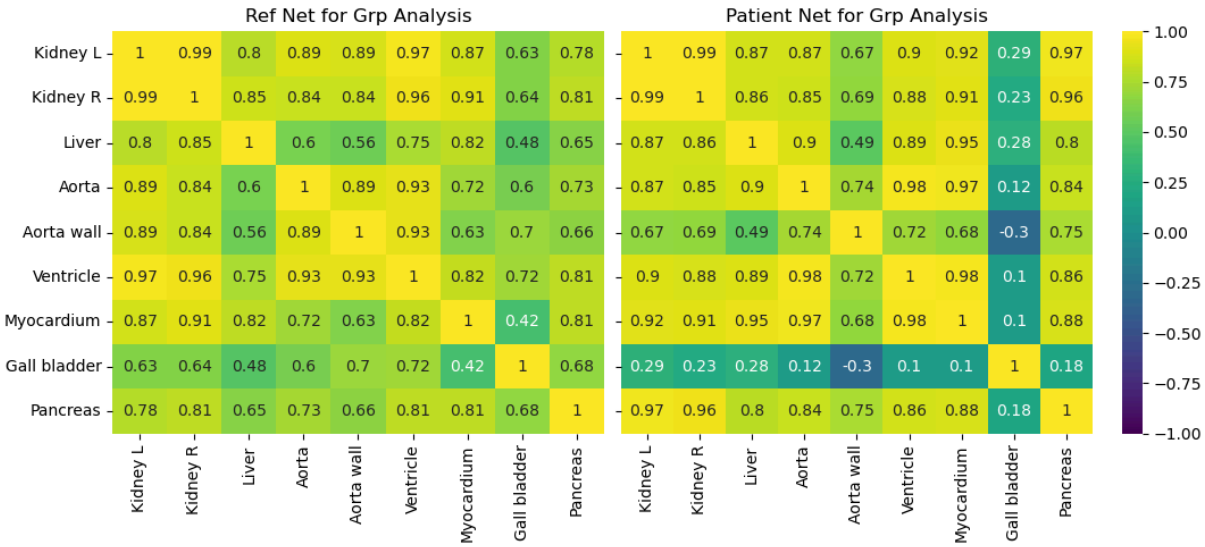


In [7]:
```
fig, ax = plt.subplots(ncols=2, figsize=(12,7),sharey=True, gridspec_kw={'wi
fig.suptitle('Individual vs. Group Z Score Analysis', fontsize=16)

#individual-level analysis
zscore_list = []
resnet_list = []
for subject in rifsubs:
    ptb = df.transpose()[ctrlsubs].copy()
    ptb = ptb.assign(rifsub=df.transpose()[subject])
    ptbnet = ptb.transpose().corr(method='pearson').transpose()
    resnet = ptbnet-refnet
    resnet_list.append(resnet)
    zscore = resnet/((1-refnet**2)/(len(ctrlsubs)-1))                #changi
    #zscore = (resnet-resnet.stack().mean())/resnet.stack().std()
    zscore_list.append(zscore)
avgzscore = sum(zscore_list)/len(zscore_list)
snszscore = sns.heatmap(avgzscore, annot=True, vmin=-5.5, vmax=2.5, cmap=pal
ax[0].set_title('Average of Individual Level Analysis')

#group-level analysis
rif = df.transpose()[rifsubs].copy().transpose()
patnet = rif.corr(method='pearson')
patnet = patnet.transpose()
# grpptbnet = rif.corr(method='pearson')
# grpptbnet = grpptbnet.transpose()
# grpresnet = grpptbnet-refnet
# grpzscore = (grpresnet-grpresnet.stack().mean())/grpresnet.stack().std()
# snsgrpzscore = sns.heatmap(grpzscore, annot=True, vmin=-4, vmax=1.5, cmap=
diffnet = (refnet-patnet)/(refnet+patnet)
snsgrp = sns.heatmap(diffnet, annot=True, vmin=-5.5, vmax=2.5, cmap=palette,
ax[1].set_title('Group Level Analysis')

fig.tight_layout()
plt.show()

compare = avgzscore.corrwith(diffnet, method='pearson')
```

```
print(compare)

fig, ax = plt.subplots(ncols=2, figsize=(11,5),sharey=True, gridspec_kw={'wi
snsgrp = sns.heatmap(refnet, annot=True, vmin=-1, vmax=1, cmap=palette, cbar
ax[0].set_title('Ref Net for Grp Analysis')
snsgrp = sns.heatmap(patnet, annot=True, vmin=-1, vmax=1, cmap=palette, ax=a
ax[1].set_title('Patient Net for Grp Analysis')
fig.tight_layout()
plt.show()
```

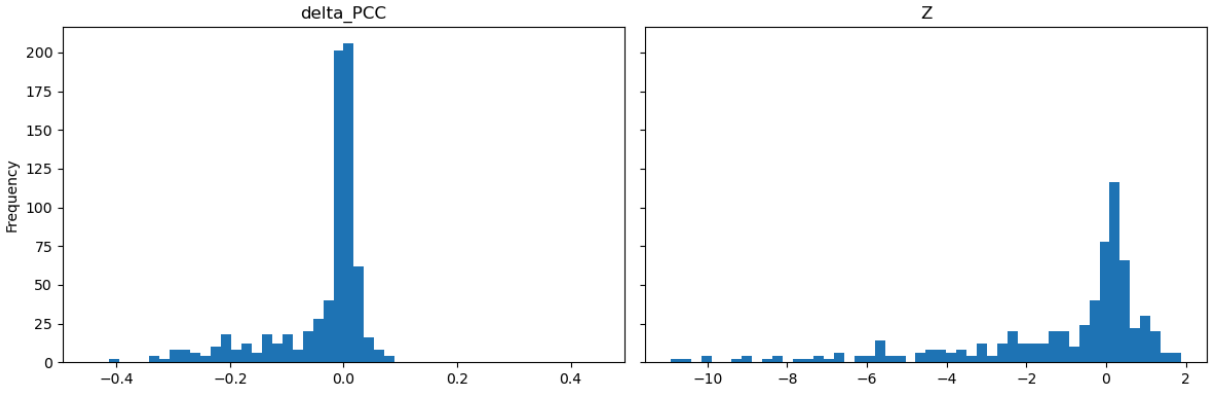



```
Kidney L       -0.042138
Kidney R        0.044509
Liver           0.673876
Aorta          -0.190313
Aorta wall     -0.478556
Ventricle      -0.190307
Myocardium     -0.030650
Gall bladder   -0.548630
Pancreas       -0.337811
dtype: float64
```

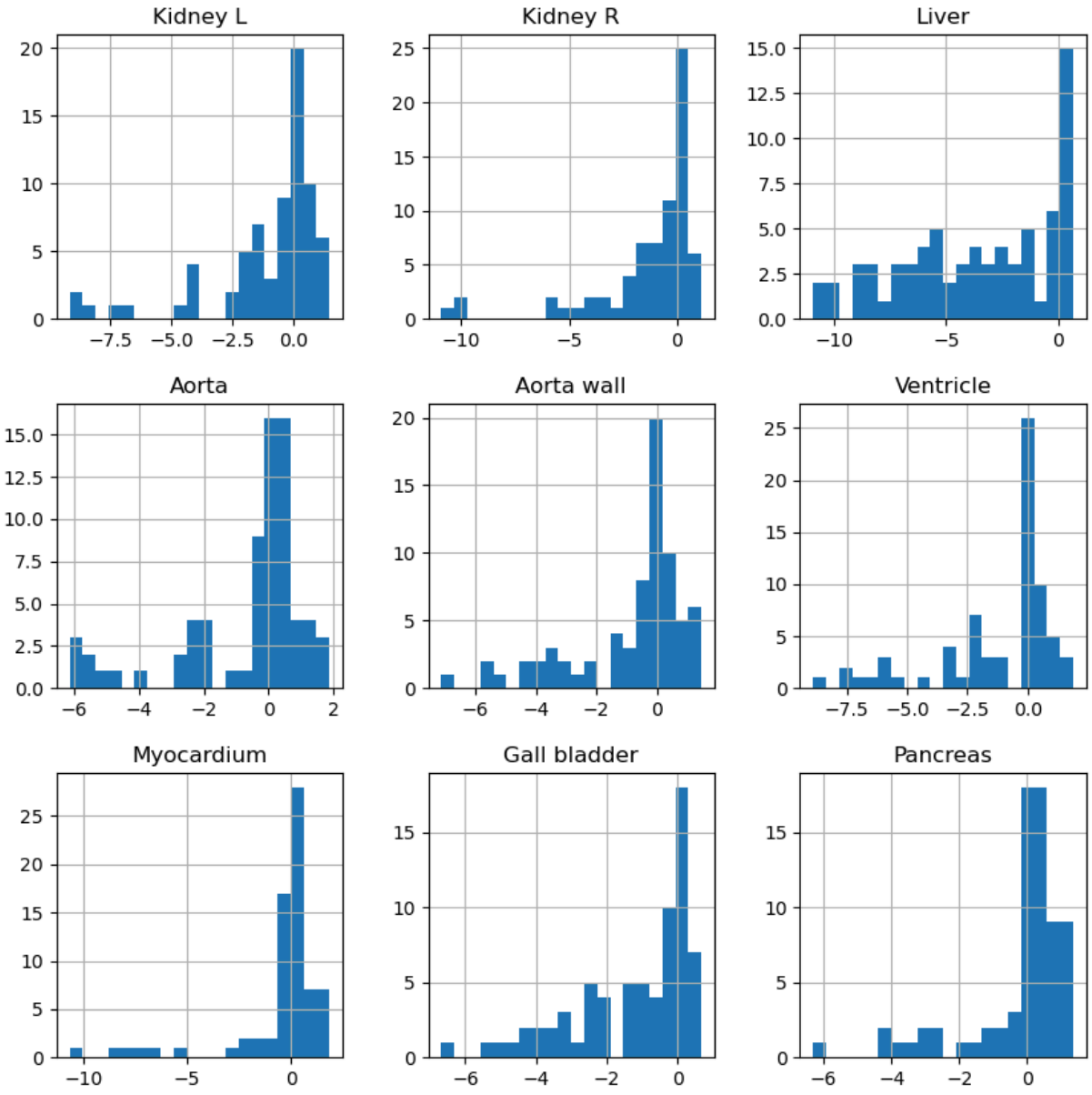


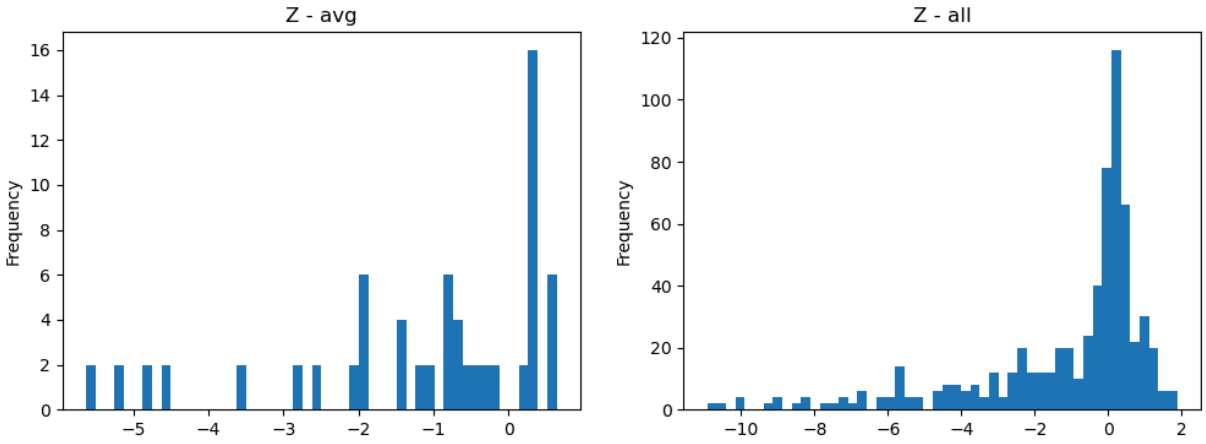
In [8]:
```
#histograms
fig, ax = plt.subplots(ncols=2, figsize=(12,4),sharey=True)
rescomb = pd.concat(resnet_list, ignore_index=True)
print("n="+str(rescomb.stack().count()))
rescomb.stack().plot.hist(bins=50, range=[-0.45, 0.45], ax=ax[0])
ax[0].set_title('delta_PCC')
zcomb = pd.concat(zscore_list, ignore_index=True)
zcomb.stack().plot.hist(bins=50, ax=ax[1])
ax[1].set_title('Z')

fig.tight_layout()
plt.show()

fig, axis = plt.subplots(3, 3, figsize=(10,10))
plt.title('Z - all')
zcomb.hist(bins=20, ax=axis)

fig, ax = plt.subplots(ncols=2, figsize=(12,4))
avgzscore.stack().plot.hist(bins=50, ax=ax[0])
ax[0].set_title('Z - avg')
zcomb.stack().plot.hist(bins=50, ax=ax[1])
ax[1].set_title('Z - all')

# fig, axis = plt.subplots(3, 3, figsize=(10,10))
# plt.title('Z - avg')
# avgzscore.hist(bins=20, range=[-3.25,3.25], ax=axis)
```

n=729



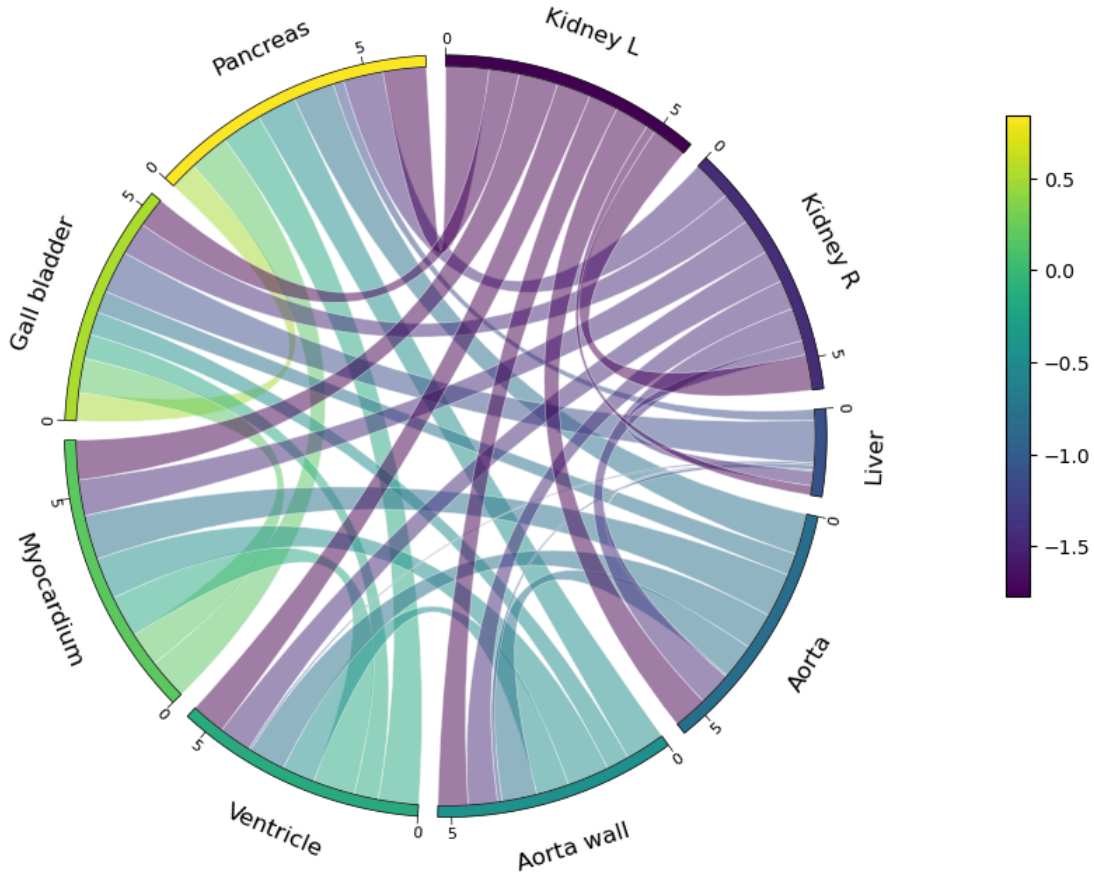
Out[8]: Text(0.5, 1.0, 'Z – all')





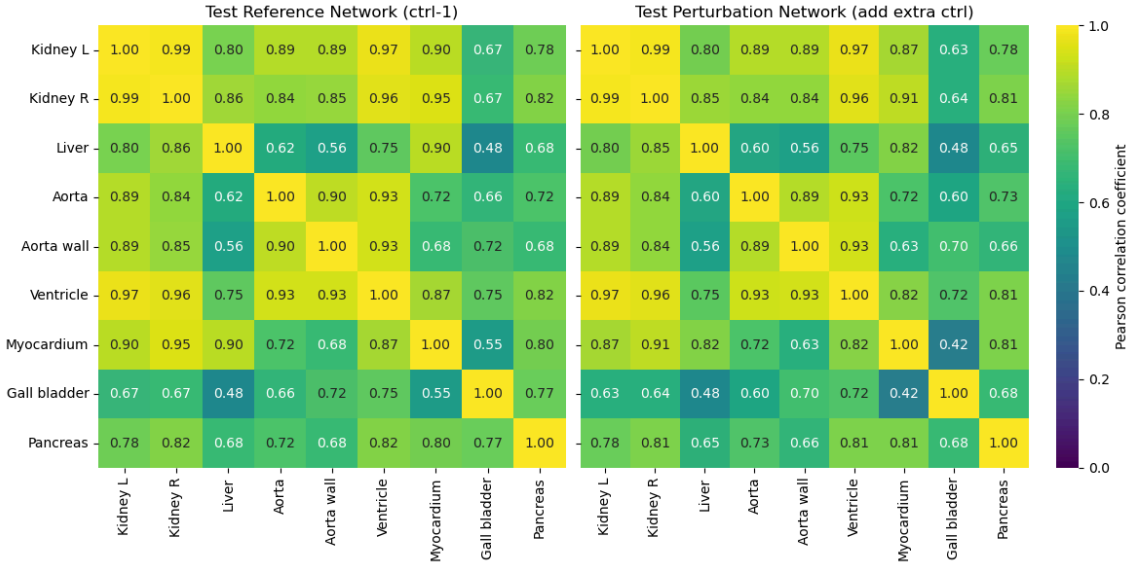
In [72]:
```
from pycirclize import Circos
from pycirclize.parser import Matrix

adjmat = nx.to_pandas_adjacency(G)
edgelist = nx.to_pandas_edgelist(G)
min=edgelist['weight'].min()
max=edgelist['weight'].max()
print('from {0:.2f} to {1:.2f}'.format(min, max))
edgelist['newweight'] = edgelist['weight']
edgelist['newweight'] += edgelist['newweight'].min()
edgelist['newweight'] -= edgelist['newweight'].min()
edgelist['newweight'] /= (edgelist['newweight'].max() - edgelist['newweight'
#print(edgelist)
edgelist['weight'] = edgelist['newweight']
edgelist.drop(columns = ['newweight'])

matrix = Matrix.parse_fromto_table(edgelist)

circos = Circos.initialize_from_matrix(
    matrix,
    space=3,
    cmap="viridis",
    ticks_interval=5,
    label_kws=dict(size=12, r=110),
    link_kws=dict(ec="white", lw=0.5),
)

circos.colorbar(cmap='viridis', vmin=min, vmax=max, bounds=(1.2, 0.3, 0.03,

fig = circos.plotfig()
```

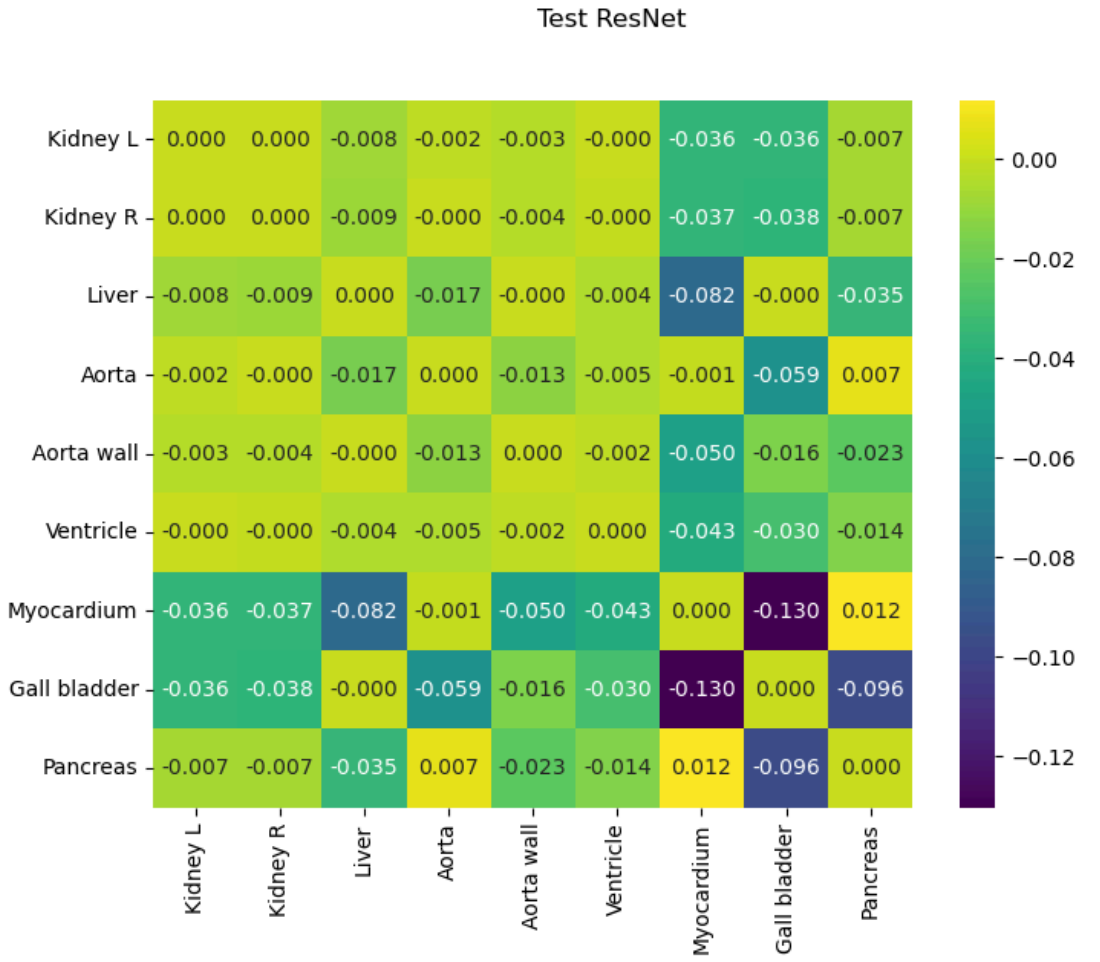
```
from -1.77 to 0.84
```

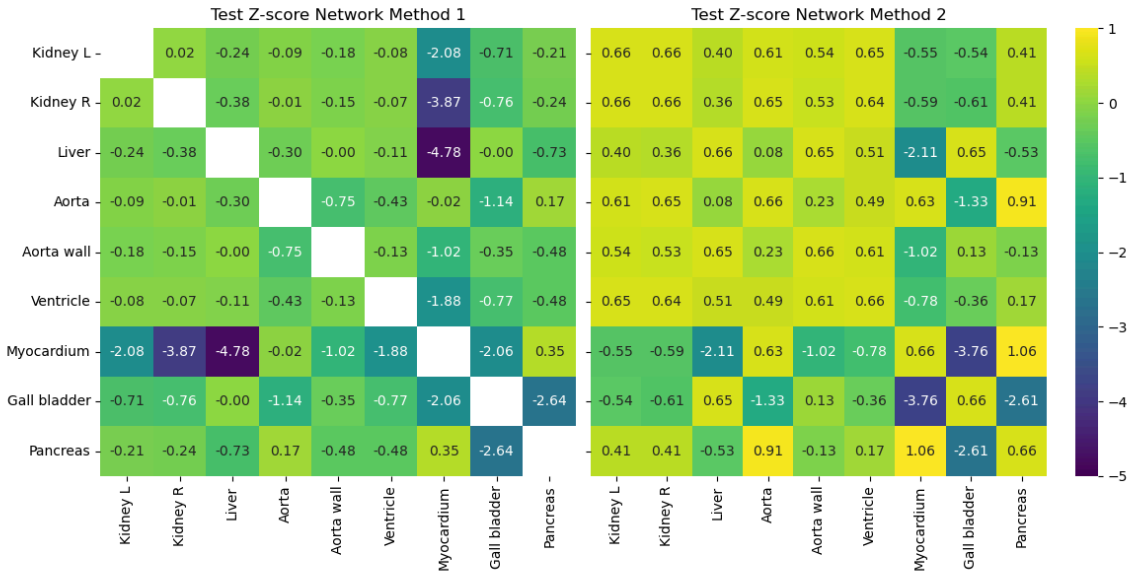


In [60]:
```
#test a control
ctrlsubs_test = ['S00732', 'S00780', 'S00786', 'S00793', 'S00796', 'S00801',
testsub = 'S00956'
ctrl_test = df.transpose()[ctrlsubs_test].copy().transpose()
refnet_test = ctrl_test.corr(method='pearson')
refnet_test = refnet_test.transpose()
fig, ax = plt.subplots(ncols=2, figsize=(12,6),sharey=True, gridspec_kw={'wi
snsctrl_test = sns.heatmap(refnet_test, annot=True, vmax=1.0, vmin=0.0, cbar
                      ax=ax[0], cmap=palette, fmt='0.2f')
ax[0].set_title('Test Reference Network (ctrl-1)')
ptb_test = df.transpose()[ctrlsubs_test].copy()
ptb_test = ptb_test.assign(testsub=df.transpose()[testsub])
ptbnet_test = ptb_test.transpose().corr(method='pearson').transpose()
snsptb_test = sns.heatmap(ptbnet_test, annot=True, vmax=1.0, vmin=0.0, cbar_
                      cmap=palette, fmt='0.2f')
ax[1].set_title('Test Perturbation Network (add extra ctrl)')
fig.tight_layout()
plt.show()
fig = plt.figure(figsize=(8,6))
resnet_test = ptbnet_test-refnet_test
snsres_test = sns.heatmap(resnet_test, annot=True, fmt='0.3f', cmap=palette)
fig.suptitle('Test ResNet')
plt.show()
fig, ax = plt.subplots(ncols=2, figsize=(12,6),sharey=True, gridspec_kw={'wi
zscore_test = resnet_test/((1-refnet_test**2)/(len(ctrlsubs_test)-1))
```

```
snszscore_test = sns.heatmap(zscore_test, annot=True, vmin=-5, vmax=1, cmap=
ax[0].set_title('Test Z-score Network Method 1')
zscore_test2 = (resnet_test-resnet_test.stack().mean())/resnet_test.stack().
snszscore_test2 = sns.heatmap(zscore_test2, annot=True, vmin=-5, vmax=1, cma
ax[1].set_title('Test Z-score Network Method 2')
fig.tight_layout()
plt.show()
```

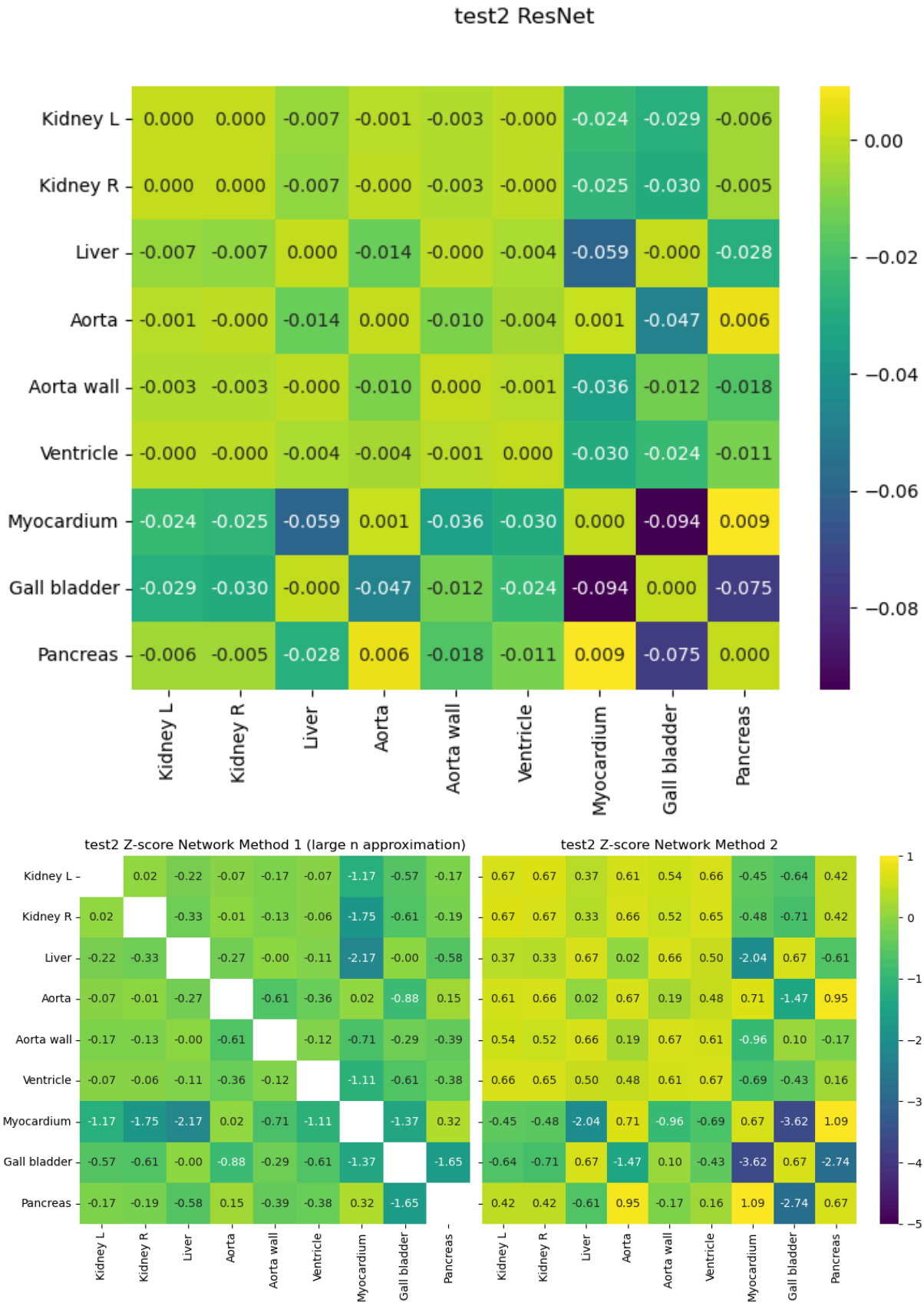






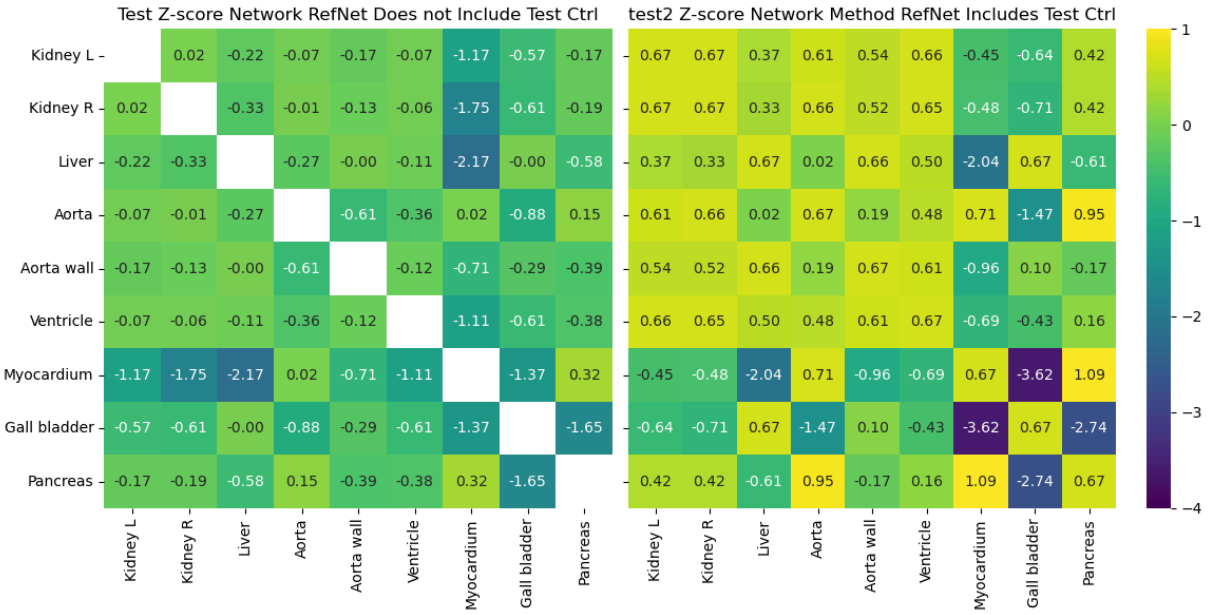
In [2]:
```
#test2 a control without removing it from ctrlsubs
ctrlsubs_test2 = ['S00732', 'S00780', 'S00786', 'S00793', 'S00796', 'S00801'
                  'S00956']
```

```
test2sub = 'S00956'
ctrl_test2 = df.transpose()[ctrlsubs_test2].copy().transpose()
refnet_test2 = ctrl_test2.corr(method='pearson')
refnet_test2 = refnet_test2.transpose()
fig, ax = plt.subplots(ncols=2, figsize=(12,6),sharey=True, gridspec_kw={'wi
snsctrl_test2 = sns.heatmap(refnet_test2, annot=True, vmax=1.0, vmin=0.0, cb
                      ax=ax[0], cmap=palette, fmt='0.2f')
ax[0].set_title('test2 Reference Network (ctrl-1)')
ptb_test2 = df.transpose()[ctrlsubs_test2].copy()
ptb_test2 = ptb_test2.assign(test2sub=df.transpose()[test2sub])
ptbnet_test2 = ptb_test2.transpose().corr(method='pearson').transpose()
snsptb_test2 = sns.heatmap(ptbnet_test2, annot=True, vmax=1.0, vmin=0.0, cba
                      cmap=palette, fmt='0.2f')
ax[1].set_title('test2 Perturbation Network (add extra ctrl)')
fig.tight_layout()
plt.show()
fig = plt.figure(figsize=(8,6))
resnet_test2 = ptbnet_test2-refnet_test2
snsres_test2 = sns.heatmap(resnet_test2, annot=True, fmt='0.3f', cmap=palett
fig.suptitle('test2 ResNet')
plt.show()
fig, ax = plt.subplots(ncols=2, figsize=(12,6),sharey=True, gridspec_kw={'wi
zscore_test2 = resnet_test2/((1-refnet_test2**2)/(len(ctrlsubs_test2)-1))
snszscore_test2 = sns.heatmap(zscore_test2, annot=True, vmin=-5, vmax=1, cma
ax[0].set_title('test2 Z-score Network Method 1 (large n approximation)')
zscore_test22 = (resnet_test2-resnet_test2.stack().mean())/resnet_test2.stac
snszscore_test22 = sns.heatmap(zscore_test22, annot=True, vmin=-5, vmax=1, c
ax[1].set_title('test2 Z-score Network Method 2')
fig.tight_layout()
plt.show()
```



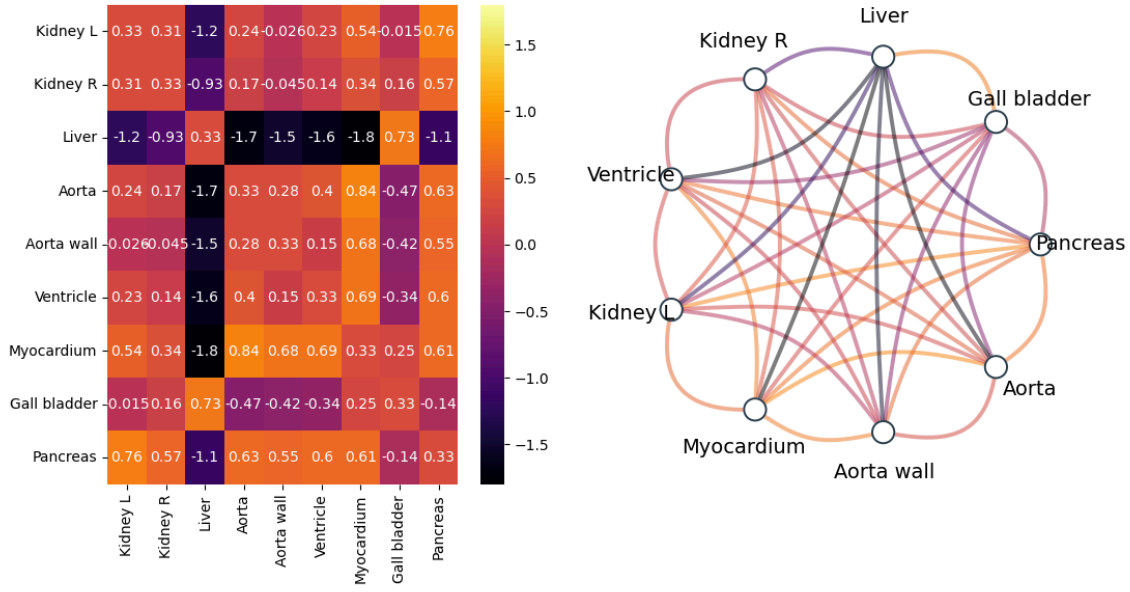
In [25]:
```
fig, ax = plt.subplots(ncols=2, figsize=(12,6),sharey=True, gridspec_kw={'wi

snszscore_test2 = sns.heatmap(zscore_test2, annot=True, vmin=-4, vmax=1, cma
ax[0].set_title('Test Z-score Network RefNet Does not Include Test Ctrl')
```

```
snszscore_test22 = sns.heatmap(zscore_test22, annot=True, vmin=-4, vmax=1, c
ax[1].set_title('test2 Z-score Network Method RefNet Includes Test Ctrl')

fig.tight_layout()
plt.show()
```

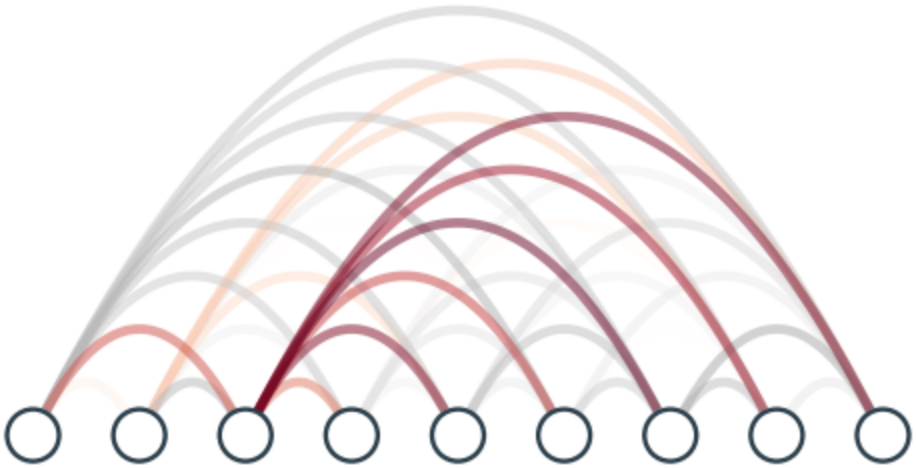


In [52]:
```
from netgraph import ArcDiagram

G = nx.from_pandas_adjacency(avgzscore, create_using=nx.Graph)
G.remove_edges_from(nx.selfloop_edges(G))
edges, weights = zip(*nx.get_edge_attributes(G, 'weight').items())

#max=zscore.stack().max()
#nx.draw_circular(G, edge_color=weights, edge_cmap=plt.get_cmap(palette), ed

fig, ax = plt.subplots(ncols=2, figsize=(12,6), gridspec_kw={'width_ratios':
snszscore = sns.heatmap(avgzscore, annot=True, vmin=-1.8, vmax=1.8, cmap='ir
Graph(G, node_labels=True, node_label_fontdict=dict(size=14), node_label_off
plt.show()

ArcDiagram(G)
```

```
/Users/abbyhellman/anaconda3/lib/python3.8/site-packages/netgraph/_utils.py:
360: RuntimeWarning: invalid value encountered in divide
  v = v / np.linalg.norm(v, axis=-1)[:, None] # unit vector
```



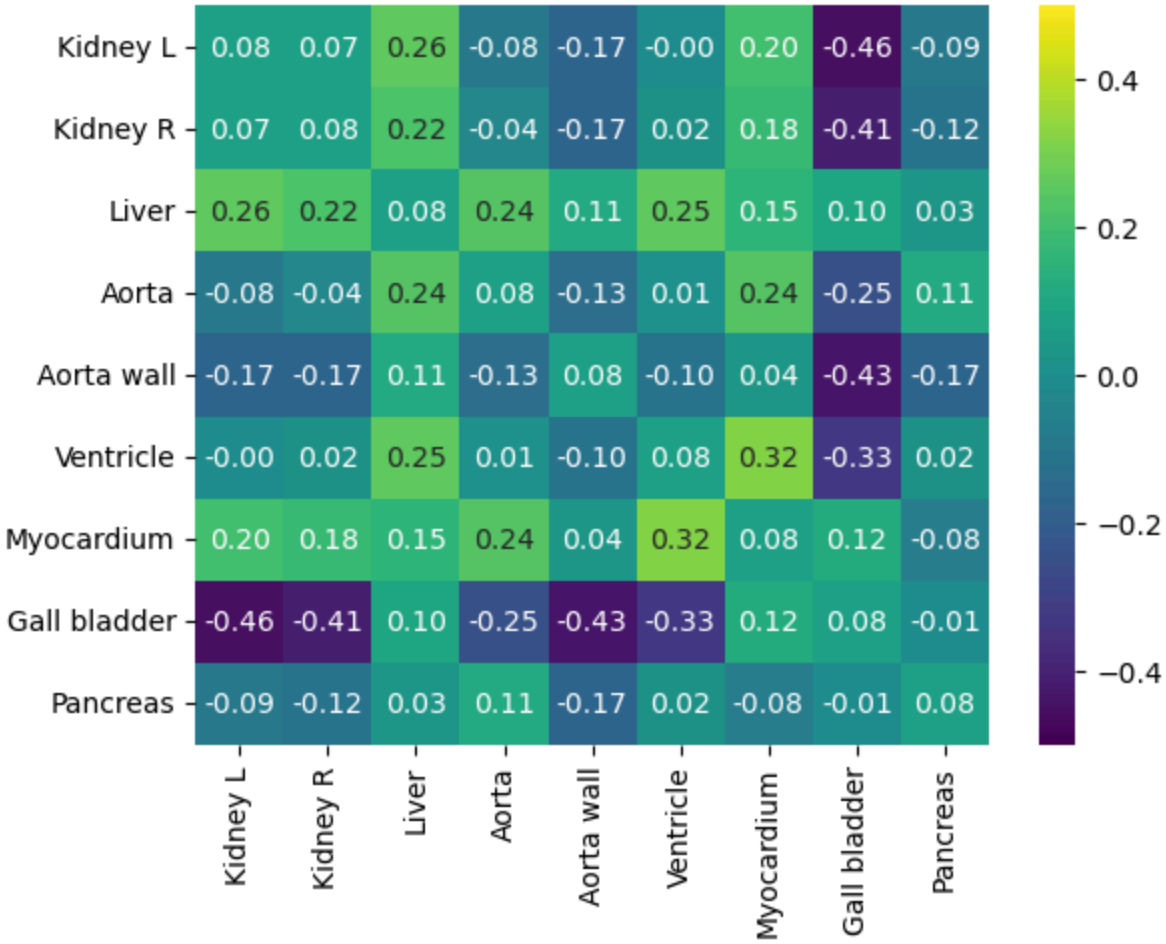
Out[52]: <netgraph._arcdiagram.ArcDiagram at 0x7ff62608a8b0>



In [6]:
```
#test every control without removing it from ctrlsubs
ctrlsubs_alltest = ['S00732', 'S00780', 'S00786', 'S00793', 'S00796', 'S0080
                'S00956']
ctrl_alltest = df.transpose()[ctrlsubs_alltest].copy().transpose()
refnet_alltest = ctrl_alltest.corr(method='pearson')
refnet_alltest = refnet_alltest.transpose()
zscore_test_list = []

for i in range(len(ctrlsubs_alltest)-1):
    testsub = ctrlsubs_alltest[i]
    ptb_alltest = df.transpose()[ctrlsubs_alltest].copy()
    ptb_alltest = ptb_alltest.assign(testsub=df.transpose()[testsub])
```

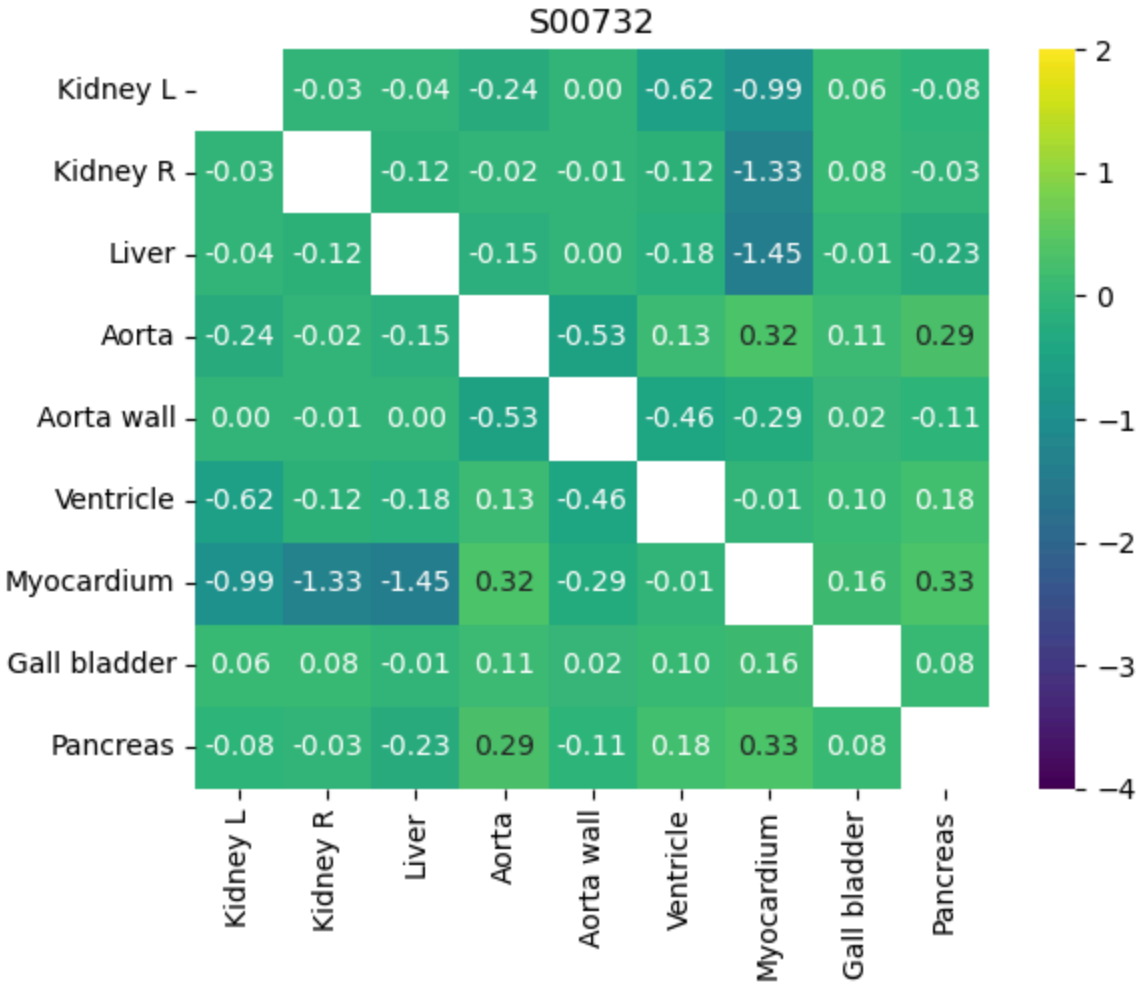
```
    ptbnet_alltest = ptb_alltest.transpose().corr(method='pearson').transpos
    resnet_alltest = ptbnet_alltest-refnet_alltest
    zscore_alltest = (resnet_alltest-resnet_alltest.stack().mean())/resnet_a
    zscore_test_list.append(zscore_alltest)

avgzscore_test = sum(zscore_test_list)/len(zscore_test_list)
print(avgzscore_test.stack().min(), avgzscore_test.stack().max())
snszscore_test = sns.heatmap(avgzscore_test, annot=True, vmin=-0.5, vmax=0.5
```

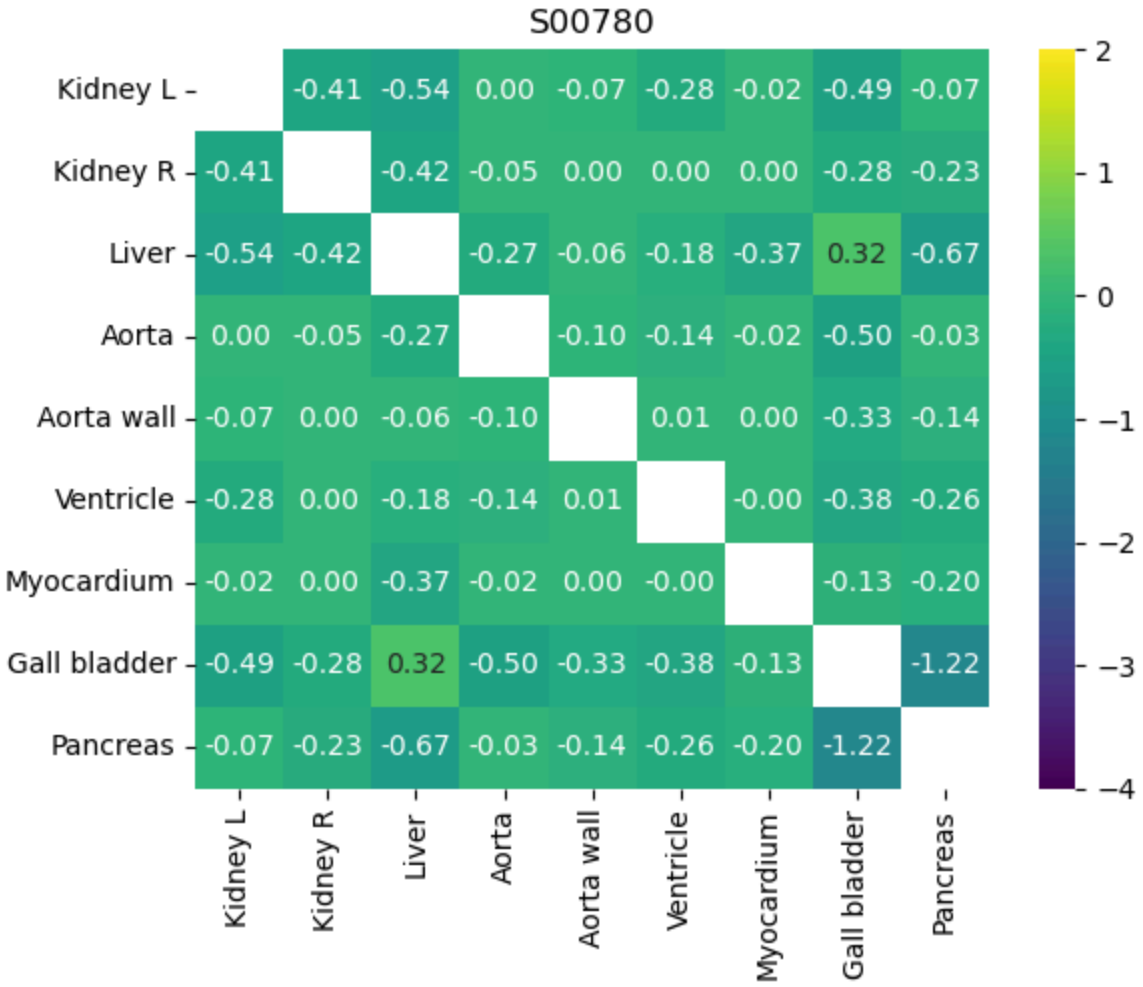
```
-0.4591038669662131 0.31740948204208513
```



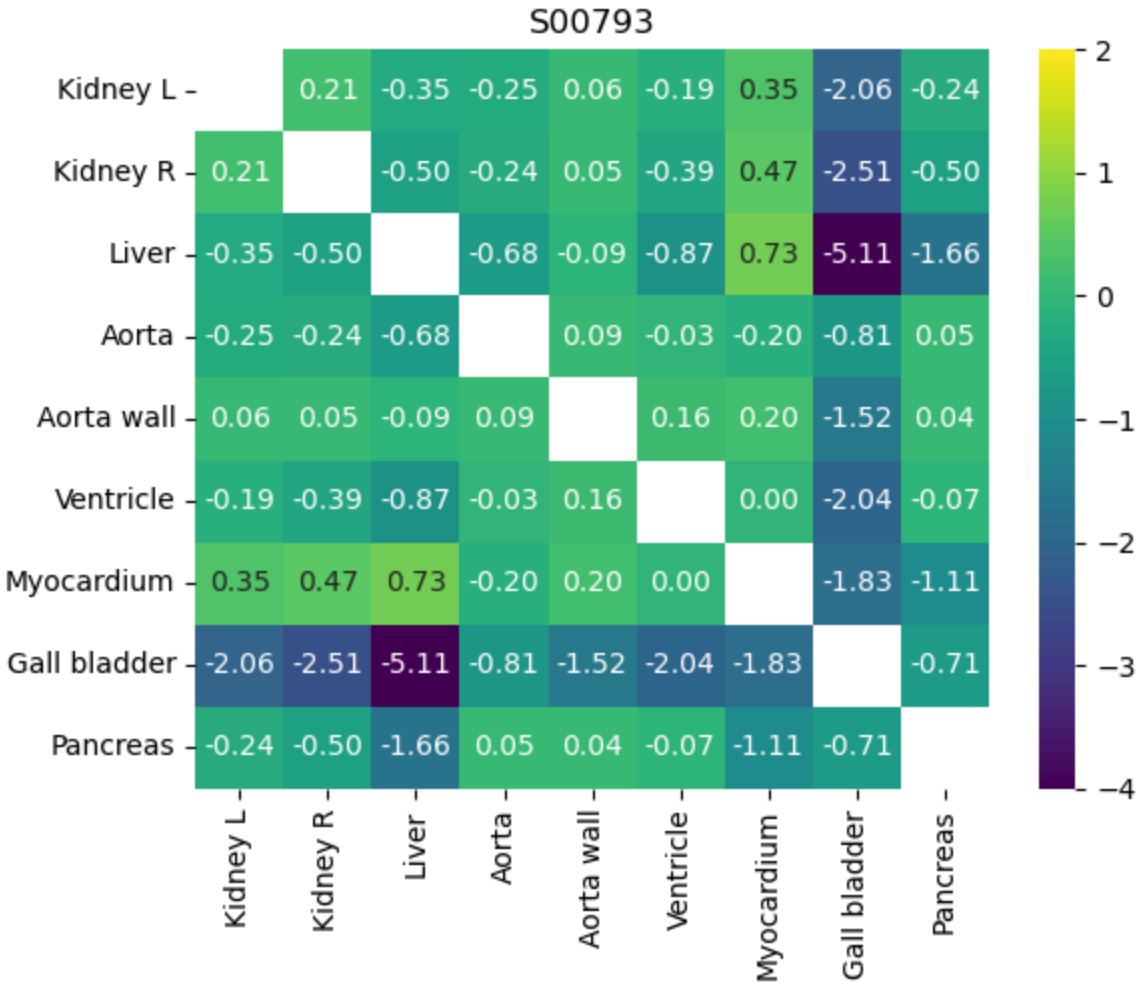
In [4]:
```
#Leave one out approach for control subjects

# resnet_list = []
zscore_list = []
for i in ctrlsubs:
    testsub = i
    ctrlsubs_l1o = ctrlsubs.copy()
    ctrlsubs_l1o.remove(testsub)
    ctrl_l1o = df.transpose()[ctrlsubs_l1o].copy().transpose()
    refnet_l1o = ctrl_l1o.corr(method='pearson')
    refnet_l1o = refnet_l1o.transpose()
    ptb_l1o = df.transpose()[ctrlsubs_l1o].copy()
    ptb_l1o = ptb_l1o.assign(testsubdf=df.transpose()[testsub])
    ptbnet_l1o = ptb_l1o.transpose().corr(method='pearson').transpose()
    resnet_l1o = ptbnet_l1o-refnet_l1o
    # resnet_list.append(resnet_l1o)
    zscore_l1o = resnet_l1o/((1-refnet_l1o**2)/(len(ctrlsubs_l1o)-1))
```

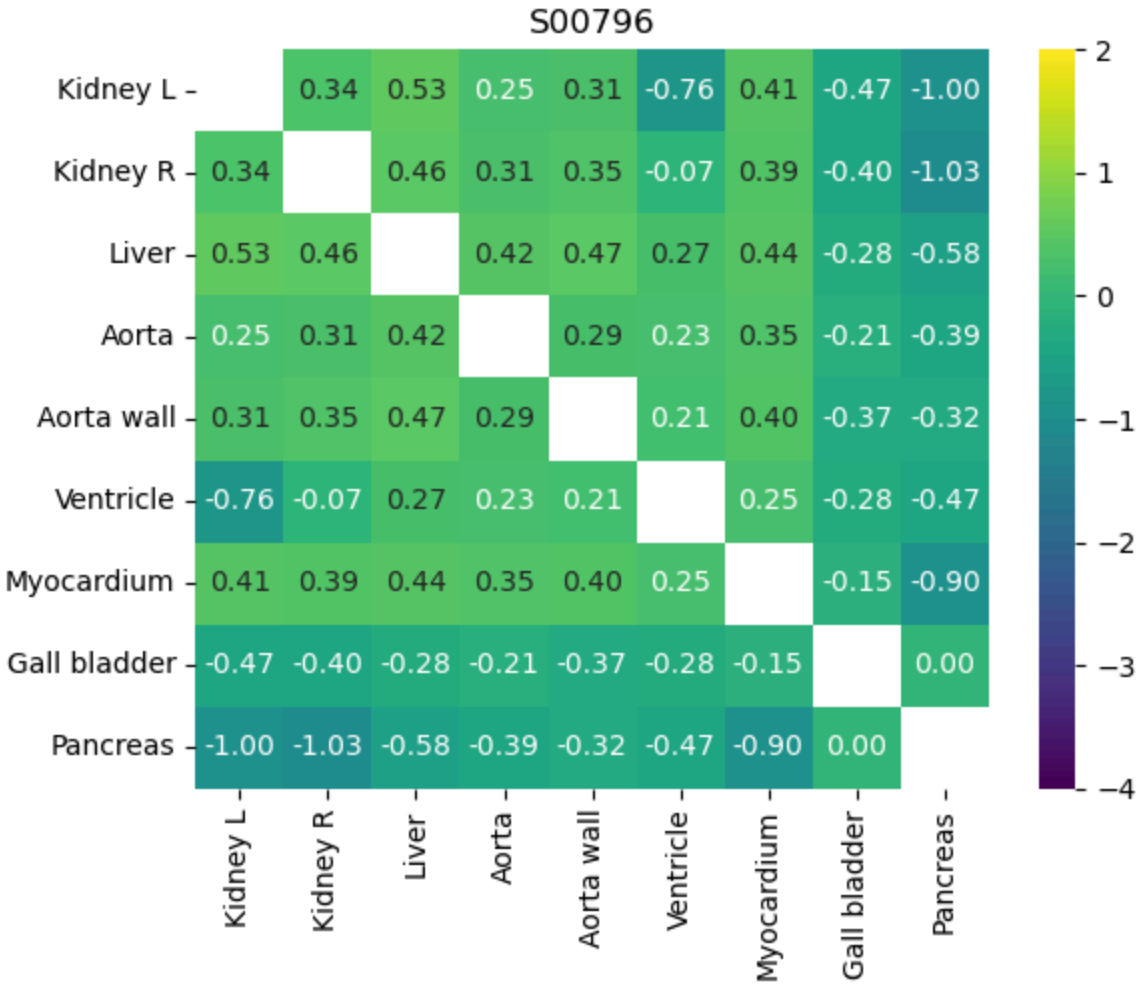
```
    # Method to plot each individual z score map
    snszscore_indiv = sns.heatmap(zscore_l1o, annot=True, vmin=-4, vmax=2, c
    plt.title(testsub)
    plt.show()
    zscore_list.append(zscore_l1o)
    # snstest = sns.heatmap(refnet_l1o, annot=True, vmin=-0.5, vmax=0.5, cma
    # plt.show()

avgzscore_l1o = sum(zscore_list)/len(zscore_list)
print('min = ', avgzscore_l1o.stack().min(), ', max = ', avgzscore_l1o.stack
snszscore_l1o = sns.heatmap(avgzscore_l1o, annot=True, vmin=-1.5, vmax=0.5,
plt.title('Average Z Score for Controls, Leave 1 Out Method')
plt.show()
avgzscore_l1o.stack().plot.hist(bins=20)
plt.title('Average Z Score Histogram')
plt.show()
# fig, axis = plt.subplots(3, 3, figsize=(10,10))
# avgzscore_l1o.hist(ax=axis)
# plt.show()

zcomb_l1o = pd.concat(zscore_list, ignore_index=True)
zcomb_l1o.stack().plot.hist(bins=50)
plt.title('Compiled Z Score Histogram')
plt.show()
fig, axis = plt.subplots(3, 3, figsize=(10,10))
zcomb_l1o.hist(ax=axis)
```

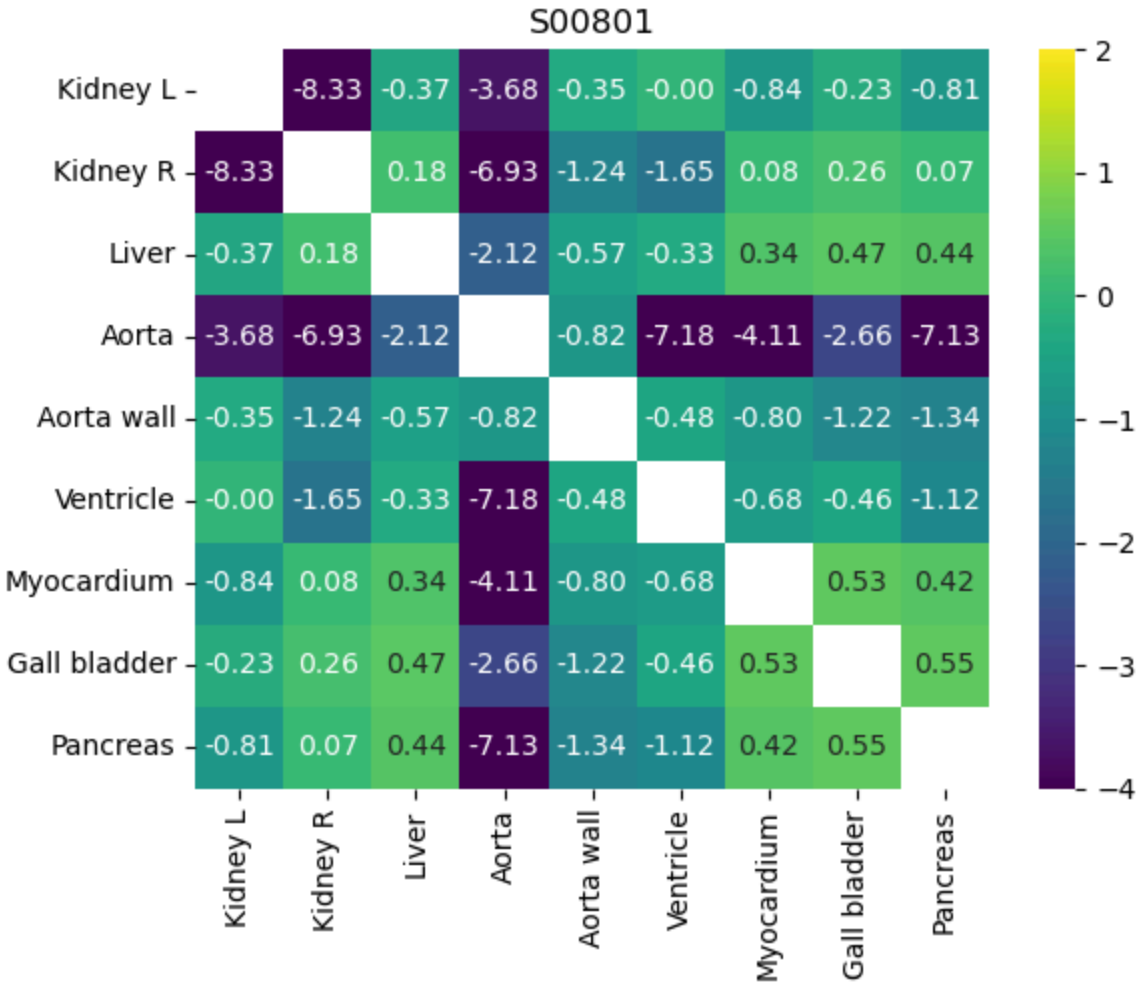

S00732
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
-0.03 -0.04 -0.24 0.00 -0.62 -0.99 0.06 -0.08
-0.03 -0.12 -0.02 -0.01 -0.12 -1.33 0.08 -0.03
-0.04 -0.12 -0.15 0.00 -0.18 -1.45 -0.01 -0.23
-0.24 -0.02 -0.15 -0.53 0.13 0.32 0.11 0.29
0.00 -0.01 0.00 -0.53 -0.46 -0.29 0.02 -0.11
-0.62 -0.12 -0.18 0.13 -0.46 -0.01 0.10 0.18
-0.99 -1.33 -1.45 0.32 -0.29 -0.01 0.16 0.33
0.06 0.08 -0.01 0.11 0.02 0.10 0.16 0.08
-0.08 -0.03 -0.23 0.29 -0.11 0.18 0.33 0.08
2
1
0
−1
−2
−3
−4

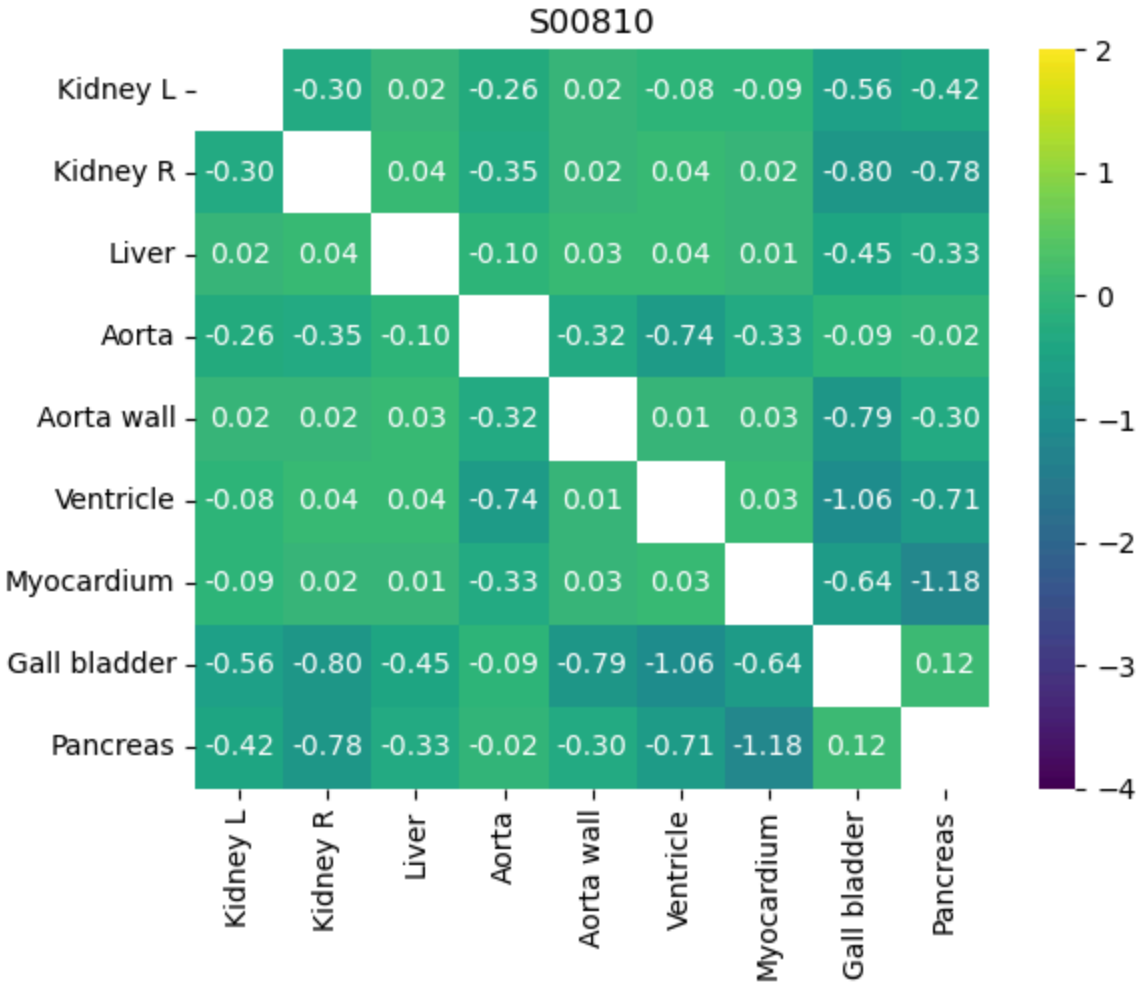

S00780
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
-0.41 -0.54 0.00 -0.07 -0.28 -0.02 -0.49 -0.07
-0.41 -0.42 -0.05 0.00 0.00 0.00 -0.28 -0.23
-0.54 -0.42 -0.27 -0.06 -0.18 -0.37 0.32 -0.67
0.00 -0.05 -0.27 -0.10 -0.14 -0.02 -0.50 -0.03
-0.07 0.00 -0.06 -0.10 0.01 0.00 -0.33 -0.14
-0.28 0.00 -0.18 -0.14 0.01 -0.00 -0.38 -0.26
-0.02 0.00 -0.37 -0.02 0.00 -0.00 -0.13 -0.20
-0.49 -0.28 0.32 -0.50 -0.33 -0.38 -0.13 -1.22
-0.07 -0.23 -0.67 -0.03 -0.14 -0.26 -0.20 -1.22
2
1
0
−1
−2
−3
−4

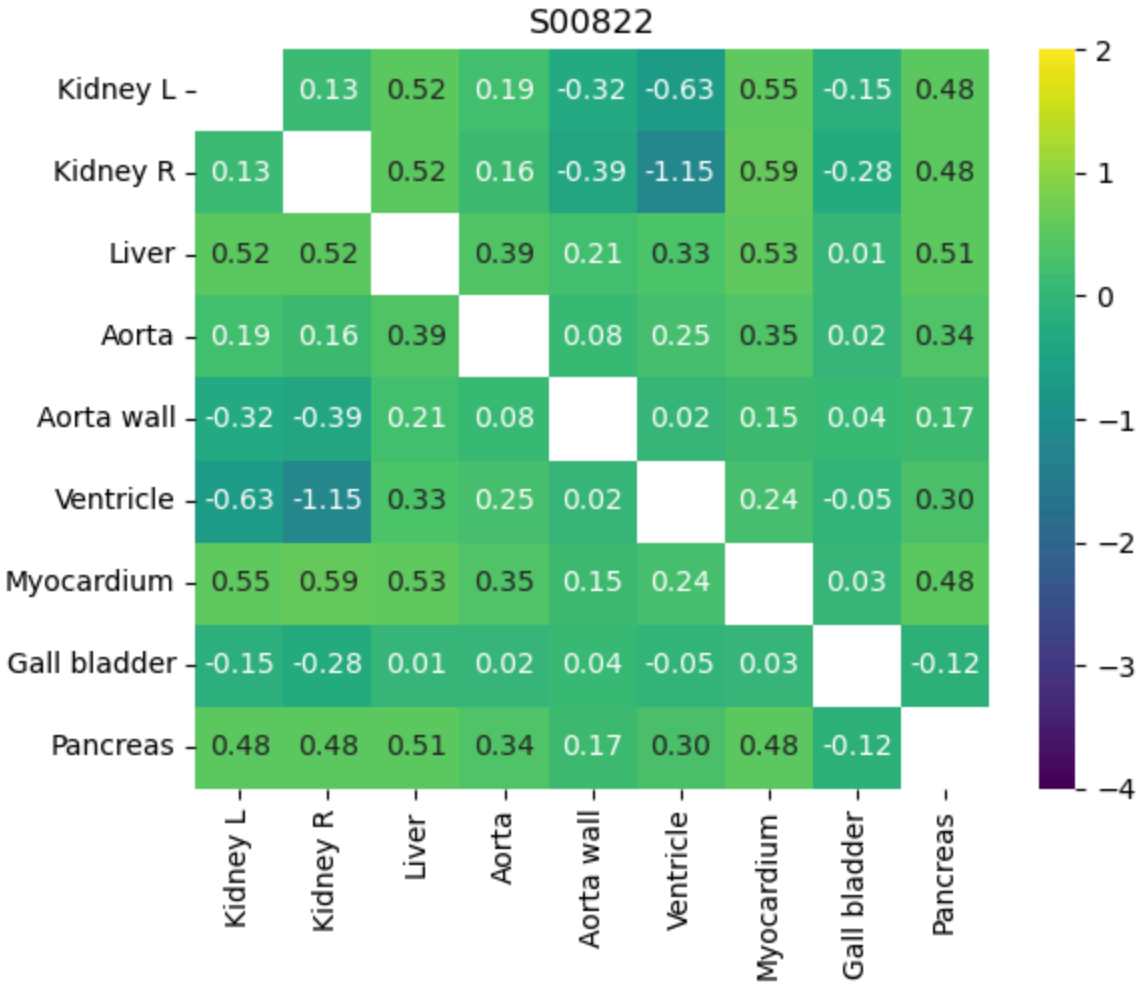

S00786
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
2.33 -2.81 4.18 4.02 2.88 -1.99 6.67 2.47
2.33 -3.11 3.99 3.46 0.35 -1.89 5.59 2.37
-2.81 -3.11 1.08 -1.27 -7.90 1.06 2.17 1.39
4.18 3.99 1.08 4.84 4.36 1.43 8.17 2.78
4.02 3.46 -1.27 4.84 5.10 0.53 10.63 3.00
2.88 0.35 -7.90 4.36 5.10 -8.29 6.64 1.26
-1.99 -1.89 1.06 1.43 0.53 -8.29 3.85 1.65
6.67 5.59 2.17 8.17 10.63 6.64 3.85 3.16
2.47 2.37 1.39 2.78 3.00 1.26 1.65 3.16
2
1
0
−1
−2
−3
−4


S00793
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
0.21 -0.35 -0.25 0.06 -0.19 0.35 -2.06 -0.24
0.21 -0.50 -0.24 0.05 -0.39 0.47 -2.51 -0.50
-0.35 -0.50 -0.68 -0.09 -0.87 0.73 -5.11 -1.66
-0.25 -0.24 -0.68 0.09 -0.03 -0.20 -0.81 0.05
0.06 0.05 -0.09 0.09 0.16 0.20 -1.52 0.04
-0.19 -0.39 -0.87 -0.03 0.16 0.00 -2.04 -0.07
0.35 0.47 0.73 -0.20 0.20 0.00 -1.83 -1.11
-2.06 -2.51 -5.11 -0.81 -1.52 -2.04 -1.83 -0.71
-0.24 -0.50 -1.66 0.05 0.04 -0.07 -1.11 -0.71
2
1
0
−1
−2
−3
−4

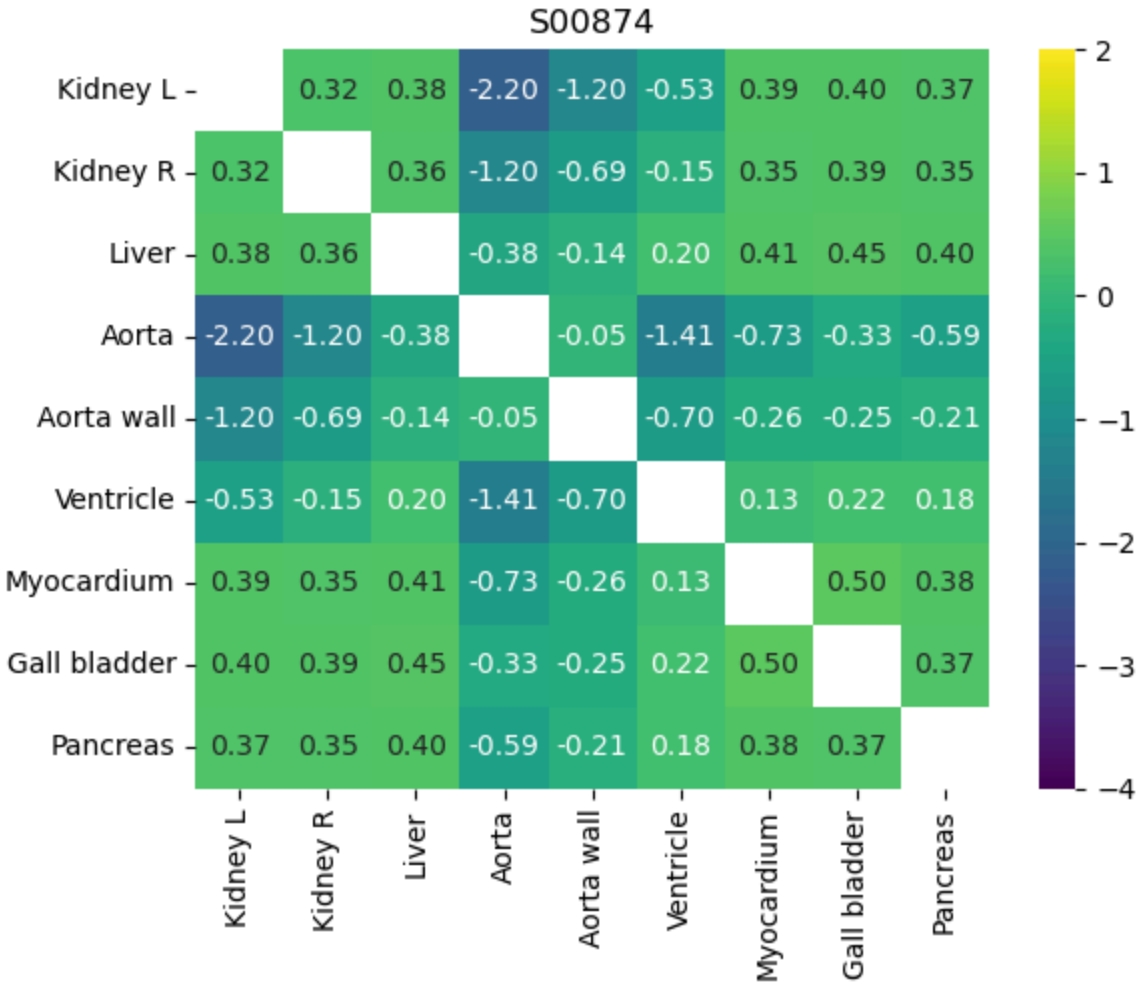

S00796
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
0.34 0.53 0.25 0.31 -0.76 0.41 -0.47 -1.00
0.34 0.46 0.31 0.35 -0.07 0.39 -0.40 -1.03
0.53 0.46 0.42 0.47 0.27 0.44 -0.28 -0.58
0.25 0.31 0.42 0.29 0.23 0.35 -0.21 -0.39
0.31 0.35 0.47 0.29 0.21 0.40 -0.37 -0.32
-0.76 -0.07 0.27 0.23 0.21 0.25 -0.28 -0.47
0.41 0.39 0.44 0.35 0.40 0.25 -0.15 -0.90
-0.47 -0.40 -0.28 -0.21 -0.37 -0.28 -0.15 0.00
-1.00 -1.03 -0.58 -0.39 -0.32 -0.47 -0.90 0.00
2
1
0
−1
−2
−3
−4

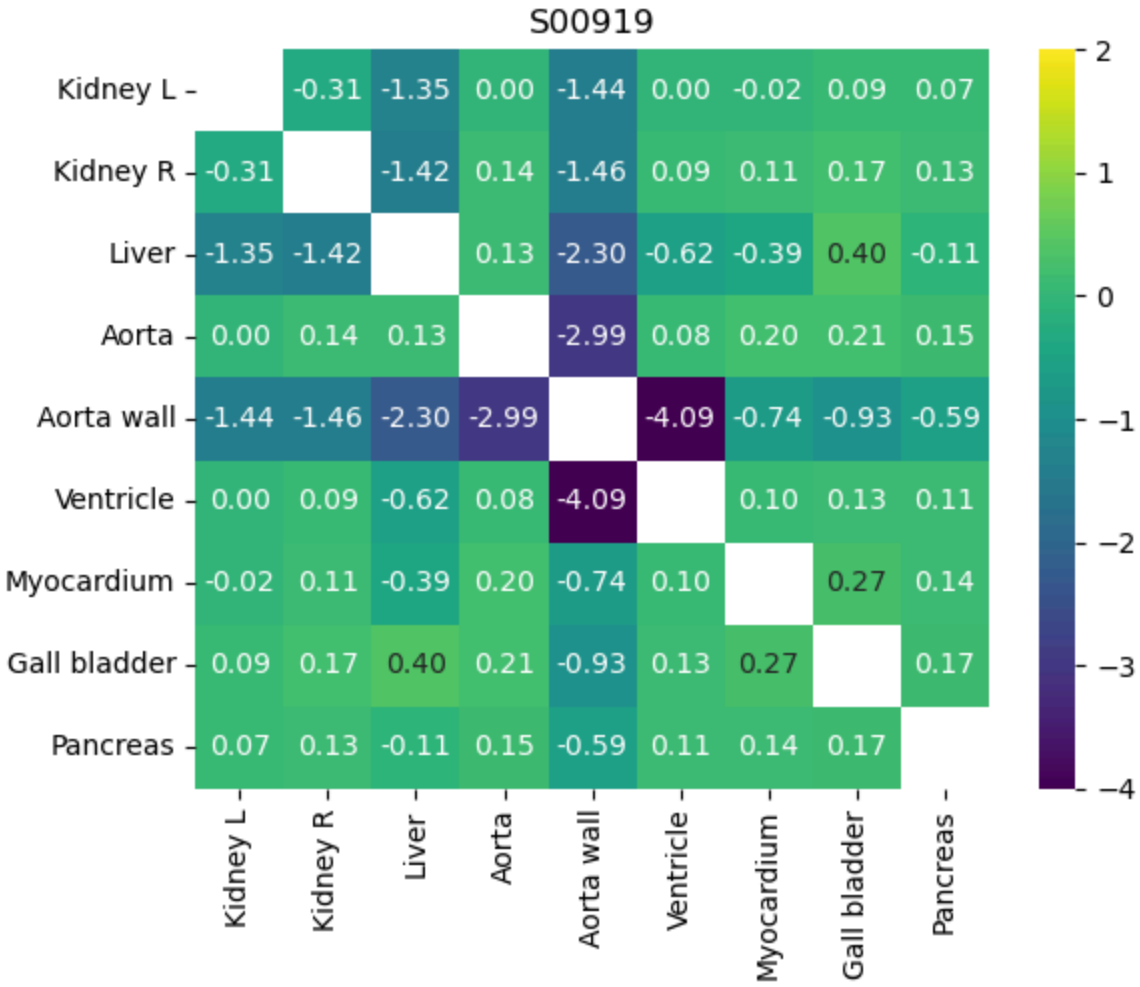

S00801
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
-8.33 -0.37 -3.68 -0.35 -0.00 -0.84 -0.23 -0.81
-8.33 0.18 -6.93 -1.24 -1.65 0.08 0.26 0.07
-0.37 0.18 -2.12 -0.57 -0.33 0.34 0.47 0.44
-3.68 -6.93 -2.12 -0.82 -7.18 -4.11 -2.66 -7.13
-0.35 -1.24 -0.57 -0.82 -0.48 -0.80 -1.22 -1.34
-0.00 -1.65 -0.33 -7.18 -0.48 -0.68 -0.46 -1.12
-0.84 0.08 0.34 -4.11 -0.80 -0.68 0.53 0.42
-0.23 0.26 0.47 -2.66 -1.22 -0.46 0.53 0.55
-0.81 0.07 0.44 -7.13 -1.34 -1.12 0.42 0.55
2
1
0
−1
−2
−3
−4

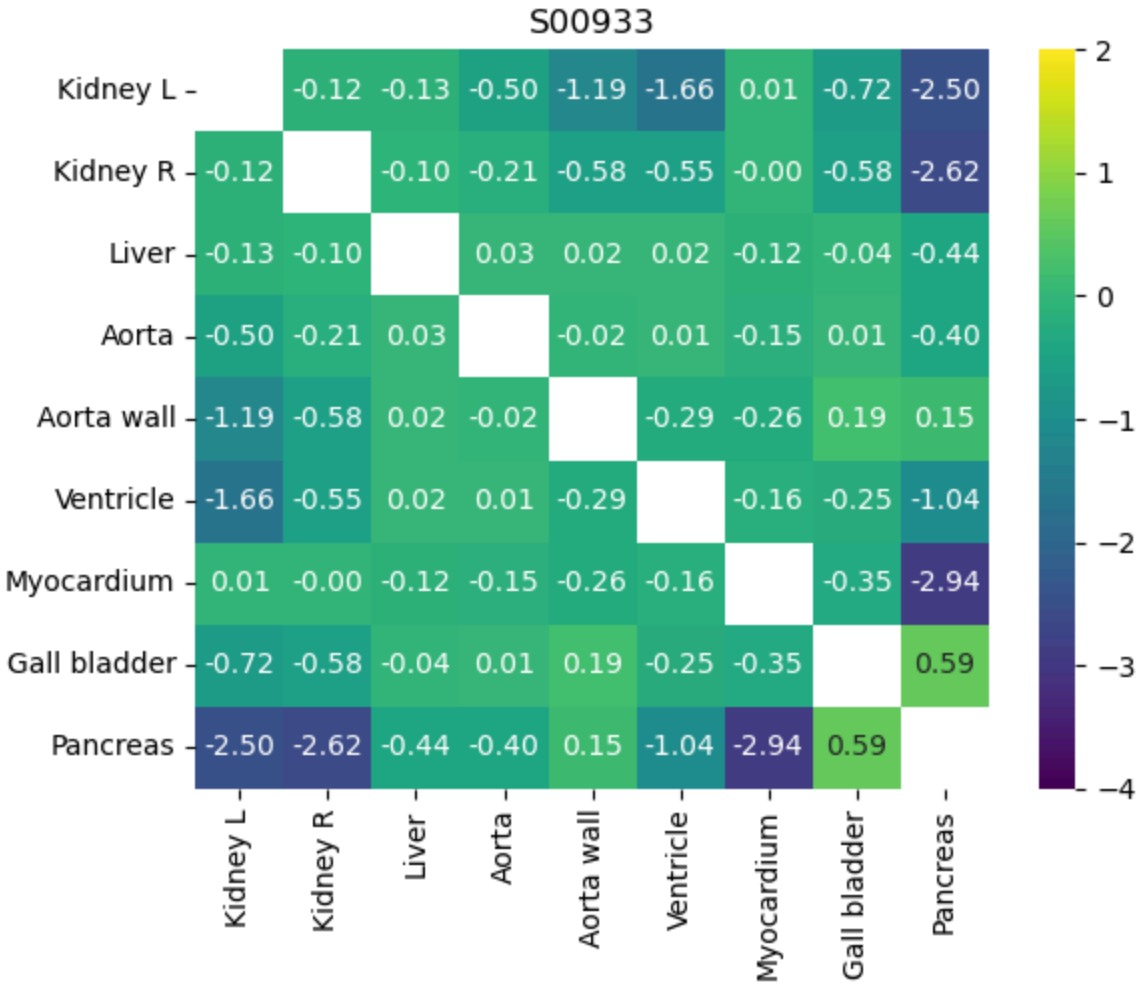

S00810
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
-0.30 0.02 -0.26 0.02 -0.08 -0.09 -0.56 -0.42
-0.30 0.04 -0.35 0.02 0.04 0.02 -0.80 -0.78
0.02 0.04 -0.10 0.03 0.04 0.01 -0.45 -0.33
-0.26 -0.35 -0.10 -0.32 -0.74 -0.33 -0.09 -0.02
0.02 0.02 0.03 -0.32 0.01 0.03 -0.79 -0.30
-0.08 0.04 0.04 -0.74 0.01 0.03 -1.06 -0.71
-0.09 0.02 0.01 -0.33 0.03 0.03 -0.64 -1.18
-0.56 -0.80 -0.45 -0.09 -0.79 -1.06 -0.64 0.12
-0.42 -0.78 -0.33 -0.02 -0.30 -0.71 -1.18 0.12
2
1
0
−1
−2
−3
−4

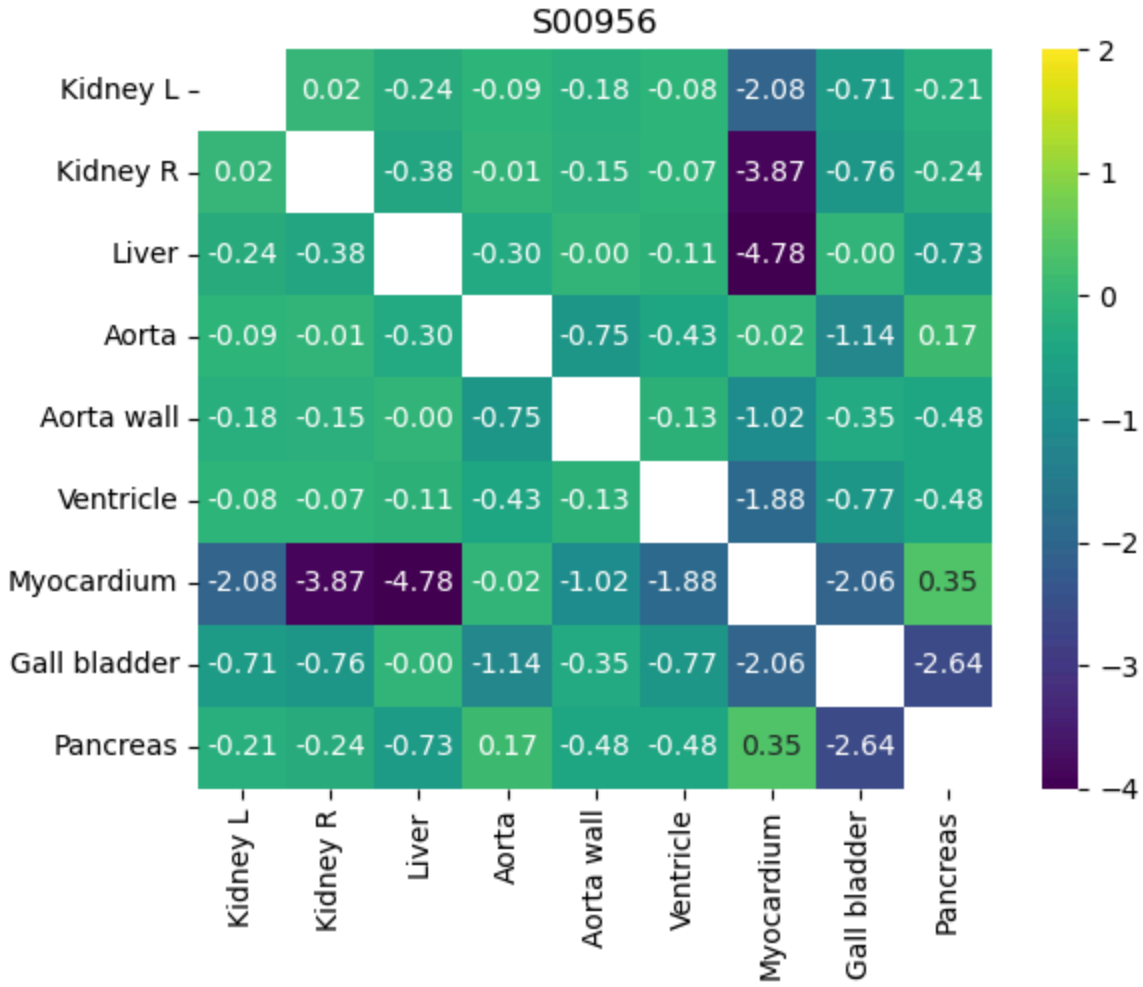

S00822
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
0.13 0.52 0.19 -0.32 -0.63 0.55 -0.15 0.48
0.13 0.52 0.16 -0.39 -1.15 0.59 -0.28 0.48
0.52 0.52 0.39 0.21 0.33 0.53 0.01 0.51
0.19 0.16 0.39 0.08 0.25 0.35 0.02 0.34
-0.32 -0.39 0.21 0.08 0.02 0.15 0.04 0.17
-0.63 -1.15 0.33 0.25 0.02 0.24 -0.05 0.30
0.55 0.59 0.53 0.35 0.15 0.24 0.03 0.48
-0.15 -0.28 0.01 0.02 0.04 -0.05 0.03 -0.12
0.48 0.48 0.51 0.34 0.17 0.30 0.48 -0.12
2
1
0
−1
−2
−3
−4


S00828
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
0.71 0.80 0.77 -0.38 0.69 0.79 -0.26 0.90
0.71 0.79 0.81 -0.16 0.69 0.79 -0.30 0.90
0.80 0.79 0.94 0.35 0.80 0.82 -0.04 1.13
0.77 0.81 0.94 -0.50 0.72 0.90 -0.23 1.05
-0.38 -0.16 0.35 -0.50 -0.88 0.27 0.03 -0.11
0.69 0.69 0.80 0.72 -0.88 0.79 -0.46 0.77
0.79 0.79 0.82 0.90 0.27 0.79 0.01 0.95
-0.26 -0.30 -0.04 -0.23 0.03 -0.46 0.01 -1.36
0.90 0.90 1.13 1.05 -0.11 0.77 0.95 -1.36
2
1
0
−1
−2
−3
−4

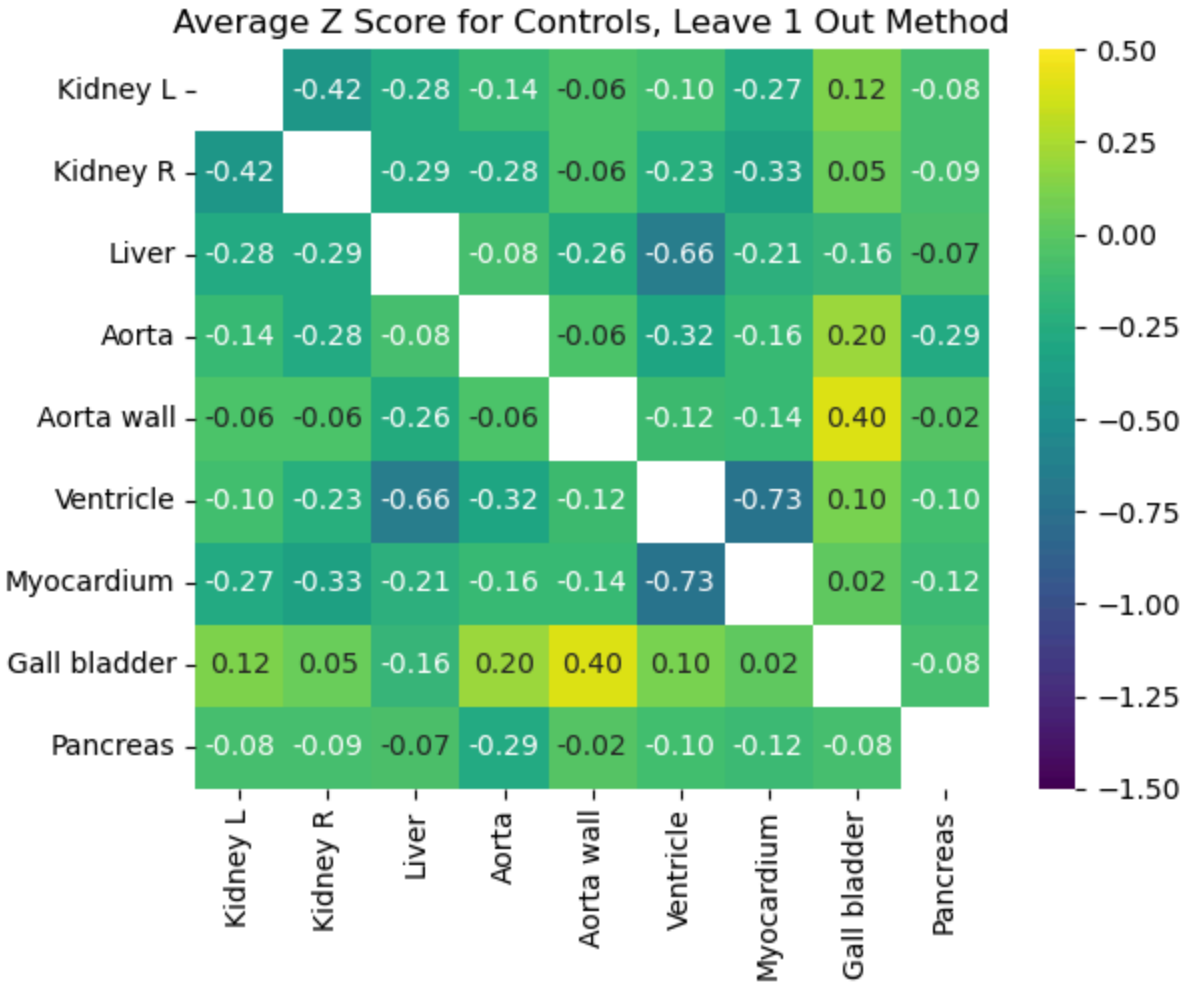

S00874
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
0.32 0.38 -2.20 -1.20 -0.53 0.39 0.40 0.37
0.32 0.36 -1.20 -0.69 -0.15 0.35 0.39 0.35
0.38 0.36 -0.38 -0.14 0.20 0.41 0.45 0.40
-2.20 -1.20 -0.38 -0.05 -1.41 -0.73 -0.33 -0.59
-1.20 -0.69 -0.14 -0.05 -0.70 -0.26 -0.25 -0.21
-0.53 -0.15 0.20 -1.41 -0.70 0.13 0.22 0.18
0.39 0.35 0.41 -0.73 -0.26 0.13 0.50 0.38
0.40 0.39 0.45 -0.33 -0.25 0.22 0.50 0.37
0.37 0.35 0.40 -0.59 -0.21 0.18 0.38 0.37
2
1
0
−1
−2
−3
−4

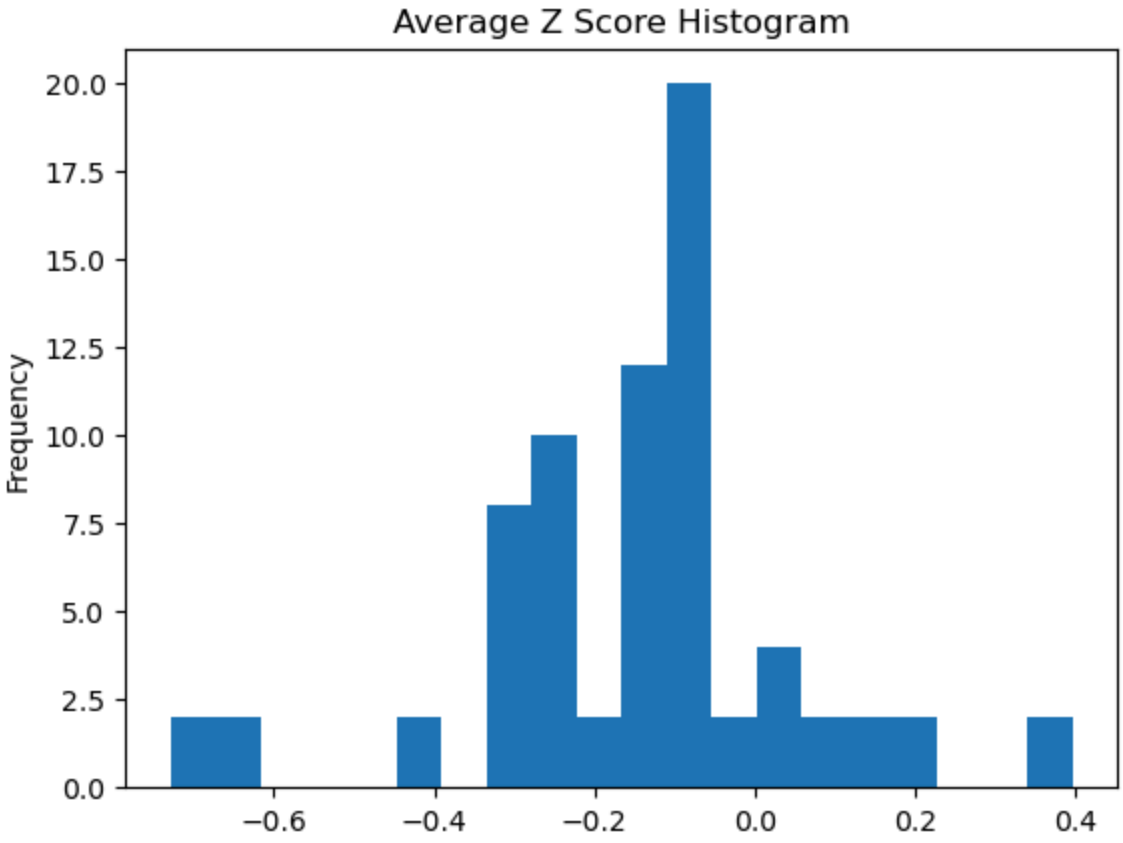

S00919
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
-0.31 -1.35 0.00 -1.44 0.00 -0.02 0.09 0.07
-0.31 -1.42 0.14 -1.46 0.09 0.11 0.17 0.13
-1.35 -1.42 0.13 -2.30 -0.62 -0.39 0.40 -0.11
0.00 0.14 0.13 -2.99 0.08 0.20 0.21 0.15
-1.44 -1.46 -2.30 -2.99 -4.09 -0.74 -0.93 -0.59
0.00 0.09 -0.62 0.08 -4.09 0.10 0.13 0.11
-0.02 0.11 -0.39 0.20 -0.74 0.10 0.27 0.14
0.09 0.17 0.40 0.21 -0.93 0.13 0.27 0.17
0.07 0.13 -0.11 0.15 -0.59 0.11 0.14 0.17
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
2
1
0
−1
−2
−3
−4

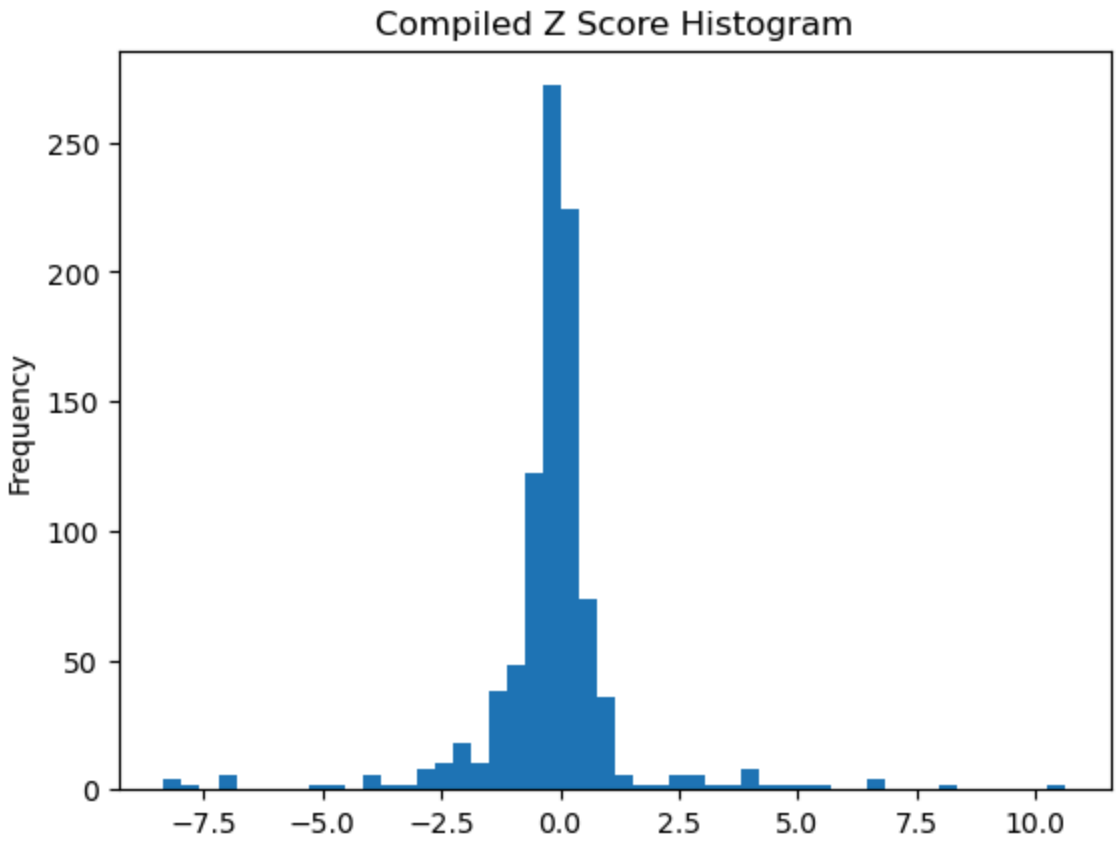

S00933
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
-0.12 -0.13 -0.50 -1.19 -1.66 0.01 -0.72 -2.50
-0.12 -0.10 -0.21 -0.58 -0.55 -0.00 -0.58 -2.62
-0.13 -0.10 0.03 0.02 0.02 -0.12 -0.04 -0.44
-0.50 -0.21 0.03 -0.02 0.01 -0.15 0.01 -0.40
-1.19 -0.58 0.02 -0.02 -0.29 -0.26 0.19 0.15
-1.66 -0.55 0.02 0.01 -0.29 -0.16 -0.25 -1.04
0.01 -0.00 -0.12 -0.15 -0.26 -0.16 -0.35 -2.94
-0.72 -0.58 -0.04 0.01 0.19 -0.25 -0.35 0.59
-2.50 -2.62 -0.44 -0.40 0.15 -1.04 -2.94 0.59
2
1
0
−1
−2
−3
−4



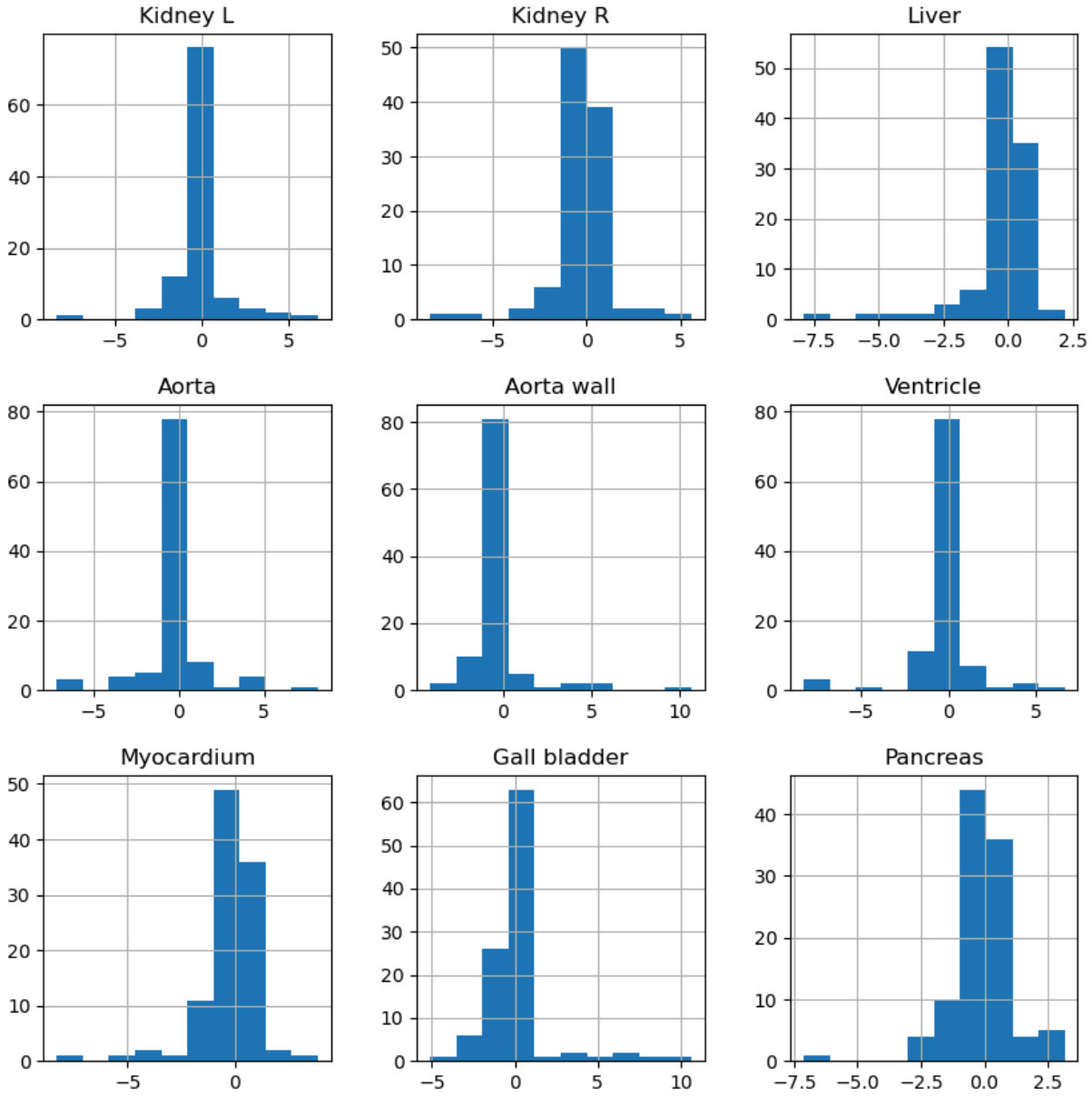

```
min =  −0.7288097899411579 , max =  0.3960518523173482
```


Average Z Score for Controls, Leave 1 Out Method
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
-0.42 -0.28 -0.14 -0.06 -0.10 -0.27 0.12 -0.08
-0.42 -0.29 -0.28 -0.06 -0.23 -0.33 0.05 -0.09
-0.28 -0.29 -0.08 -0.26 -0.66 -0.21 -0.16 -0.07
-0.14 -0.28 -0.08 -0.06 -0.32 -0.16 0.20 -0.29
-0.06 -0.06 -0.26 -0.06 -0.12 -0.14 0.40 -0.02
-0.10 -0.23 -0.66 -0.32 -0.12 -0.73 0.10 -0.10
-0.27 -0.33 -0.21 -0.16 -0.14 -0.73 0.02 -0.12
0.12 0.05 -0.16 0.20 0.40 0.10 0.02 -0.08
-0.08 -0.09 -0.07 -0.29 -0.02 -0.10 -0.12 -0.08
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.25
−1.50


Average Z Score Histogram
Frequency
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
−0.6
−0.4
−0.2
0.0
0.2
0.4

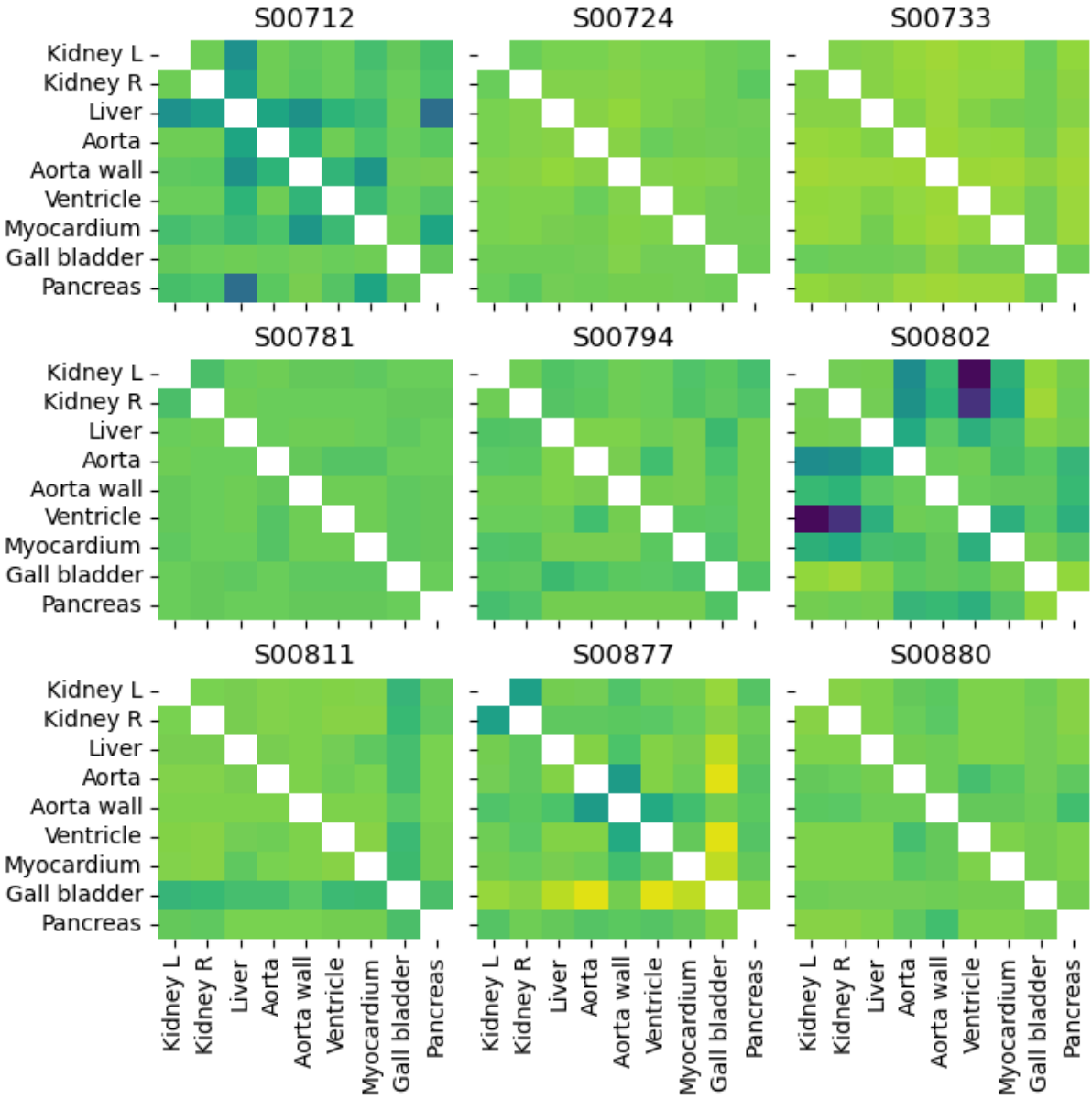


Out[4]:
```
array([[<Axes: title={'center': 'Kidney L'}>,
        <Axes: title={'center': 'Kidney R'}>,
        <Axes: title={'center': 'Liver'}>],
       [<Axes: title={'center': 'Aorta'}>,
        <Axes: title={'center': 'Aorta wall'}>,
        <Axes: title={'center': 'Ventricle'}>],
       [<Axes: title={'center': 'Myocardium'}>,
        <Axes: title={'center': 'Gall bladder'}>,
        <Axes: title={'center': 'Pancreas'}>]], dtype=object)
```

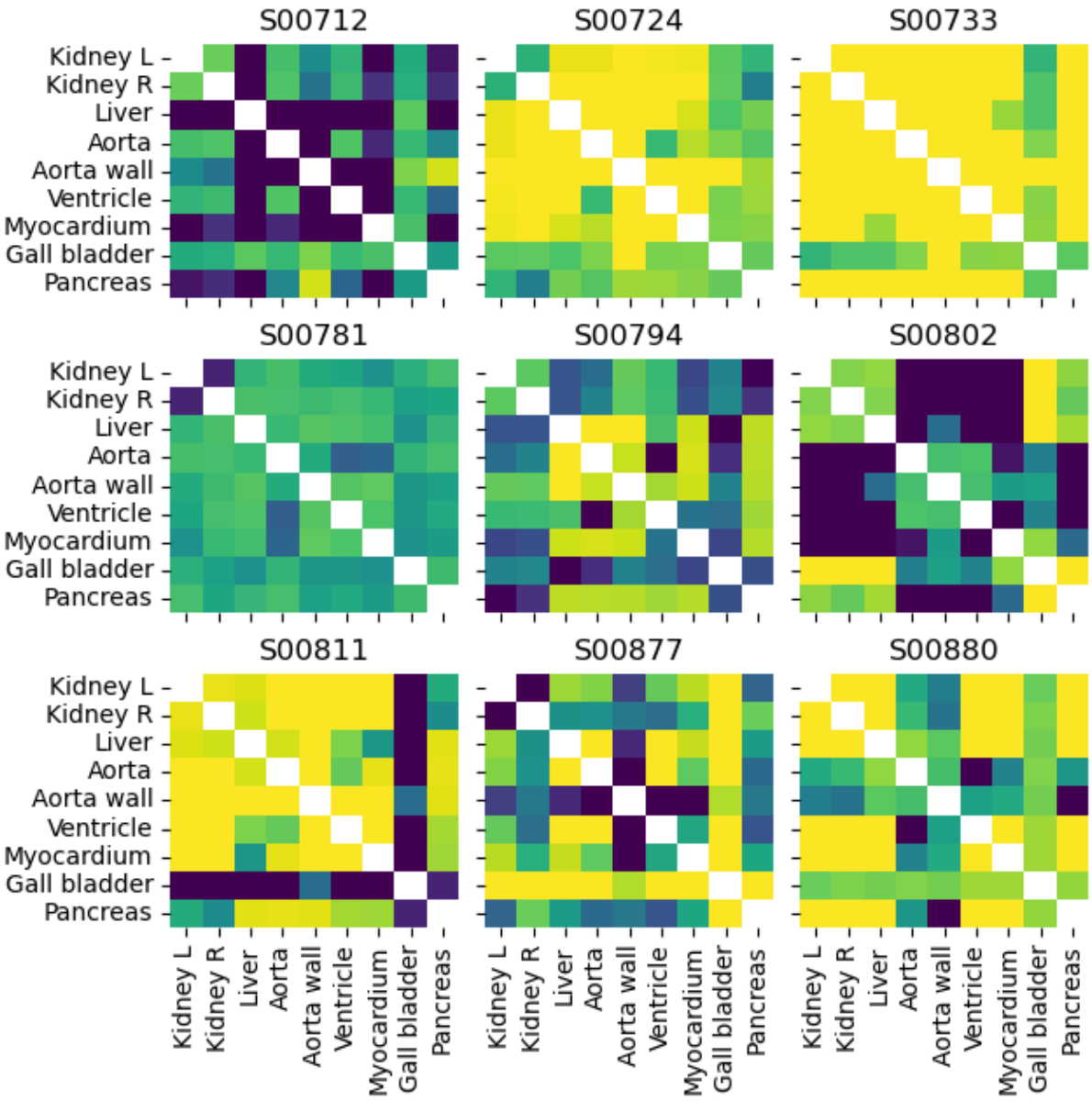


In [33]:
```
#Now the same but only the rifampicin subjects

fig, axes = plt.subplots(3, 3, sharey=True, sharex=True, figsize=(7,7))
zscore_list = []
i=0
for testsub in rifsubs:
    rifsubs_l1o = rifsubs.copy()
    rifsubs_l1o.remove(testsub)
    rif_l1o = df.transpose()[rifsubs_l1o].copy().transpose()
    refnet_l1o = rif_l1o.corr(method='pearson')
    refnet_l1o = refnet_l1o.transpose()
    ptb_l1o = df.transpose()[rifsubs_l1o].copy()
    ptb_l1o = ptb_l1o.assign(testsubdf=df.transpose()[testsub])
    ptbnet_l1o = ptb_l1o.transpose().corr(method='pearson').transpose()
    resnet_l1o = ptbnet_l1o-refnet_l1o
    # resnet_list.append(resnet_l1o)
    zscore_l1o = resnet_l1o/((1-refnet_l1o**2)/(len(rifsubs_l1o)-1))
    # Method to plot each individual z score map
    snszscore_indiv = sns.heatmap(zscore_l1o, vmin=-14, vmax=4, cmap=palette
```

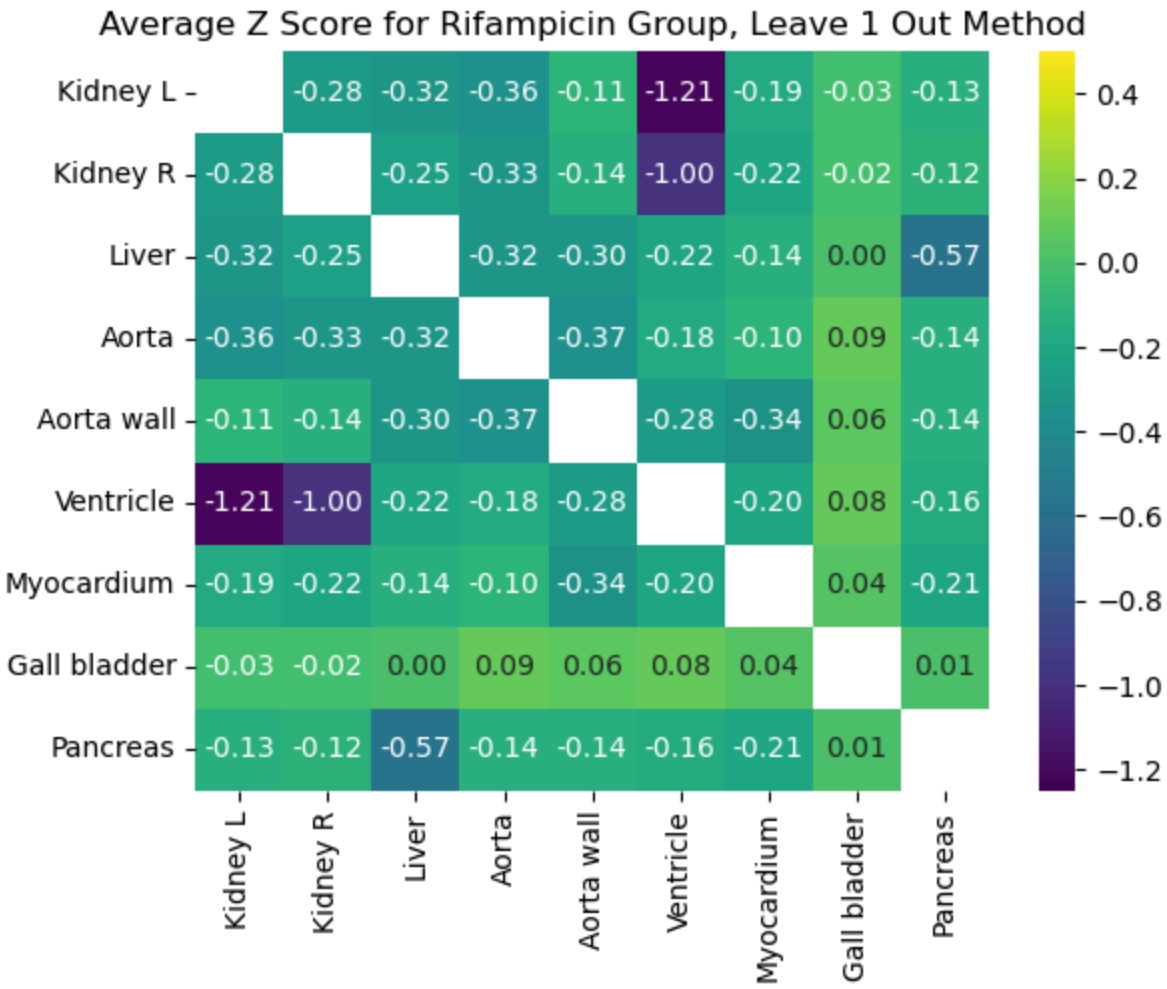
```
    axes.flat[i].set_title(testsub)
    zscore_list.append(zscore_l1o)
    i += 1
fig.tight_layout()
plt.show()
print('range = [-14, 4]')

#repeat with smaller range
fig, axes = plt.subplots(3, 3, sharey=True, sharex=True, figsize=(6.5,6.5))
zscore_list = []
i=0
for testsub in rifsubs:
    rifsubs_l1o = rifsubs.copy()
    rifsubs_l1o.remove(testsub)
    rif_l1o = df.transpose()[rifsubs_l1o].copy().transpose()
    refnet_l1o = rif_l1o.corr(method='pearson')
    refnet_l1o = refnet_l1o.transpose()
    ptb_l1o = df.transpose()[rifsubs_l1o].copy()
    ptb_l1o = ptb_l1o.assign(testsubdf=df.transpose()[testsub])
    ptbnet_l1o = ptb_l1o.transpose().corr(method='pearson').transpose()
    resnet_l1o = ptbnet_l1o-refnet_l1o
    # resnet_list.append(resnet_l1o)
    zscore_l1o = resnet_l1o/((1-refnet_l1o**2)/(len(rifsubs_l1o)-1))
    # Method to plot each individual z score map
    snszscore_indiv = sns.heatmap(zscore_l1o, vmin=-1.25, vmax=0.5, cmap=pal
    axes.flat[i].set_title(testsub)
    zscore_list.append(zscore_l1o)
    i += 1
fig.tight_layout()
plt.show()
print('range = [-1.25, 0.5]')

avgzscore_l1o = sum(zscore_list)/len(zscore_list)
# print('min = ', avgzscore_l1o.stack().min(), ', max = ', avgzscore_l1o.sta
snszscore_l1o = sns.heatmap(avgzscore_l1o, annot=True, vmin=-1.25, vmax=0.5,
plt.title('Average Z Score for Rifampicin Group, Leave 1 Out Method')
plt.show()
avgzscore_l1o.stack().plot.hist(bins=10)
plt.title('Average Z Score Histogram')
plt.show()
# fig, axis = plt.subplots(3, 3, figsize=(10,10))
# avgzscore_l1o.hist(ax=axis)
# plt.show()

zcomb_l1o = pd.concat(zscore_list, ignore_index=True)
zcomb_l1o.stack().plot.hist(bins=50)
plt.title('Compiled Z Scores Histogram')
plt.show()
fig, axis = plt.subplots(3, 3, figsize=(10,10))
zcomb_l1o.hist(ax=axis)
```

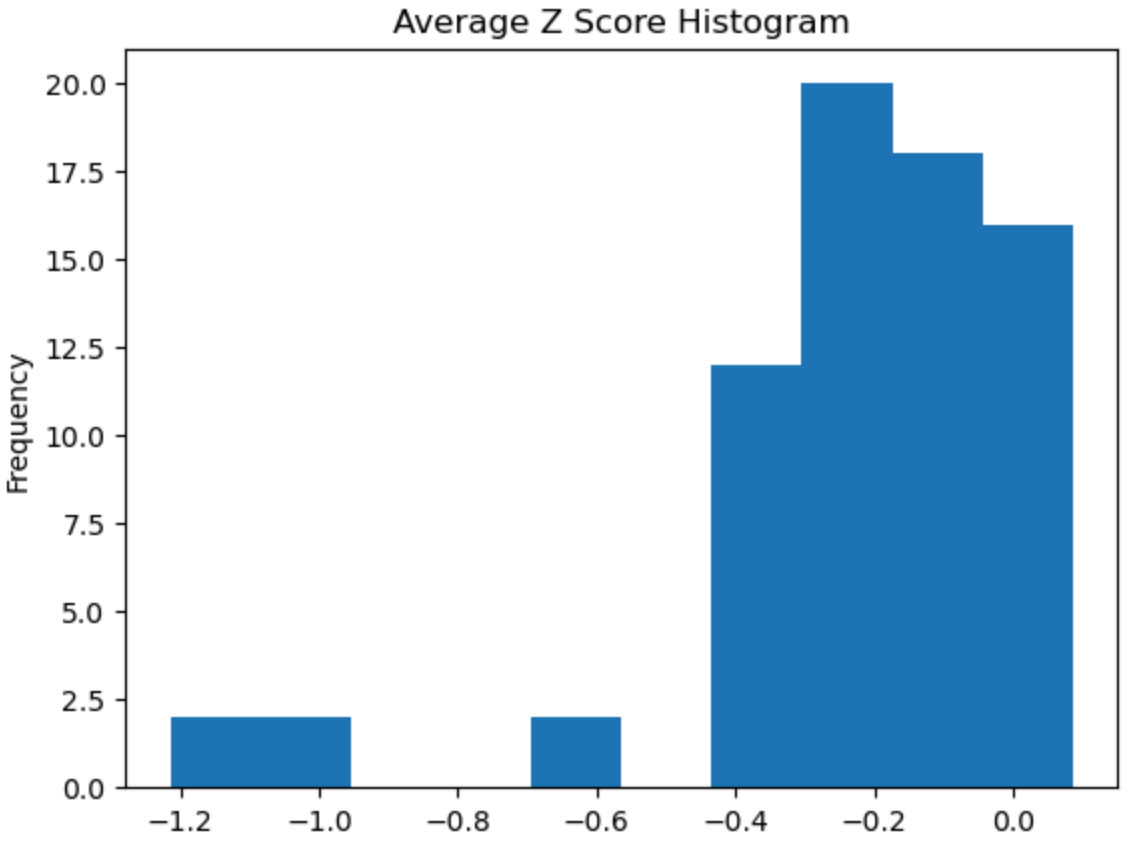

S00712
S00724
S00733
S00781
S00794
S00802
S00811
S00877
S00880
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas

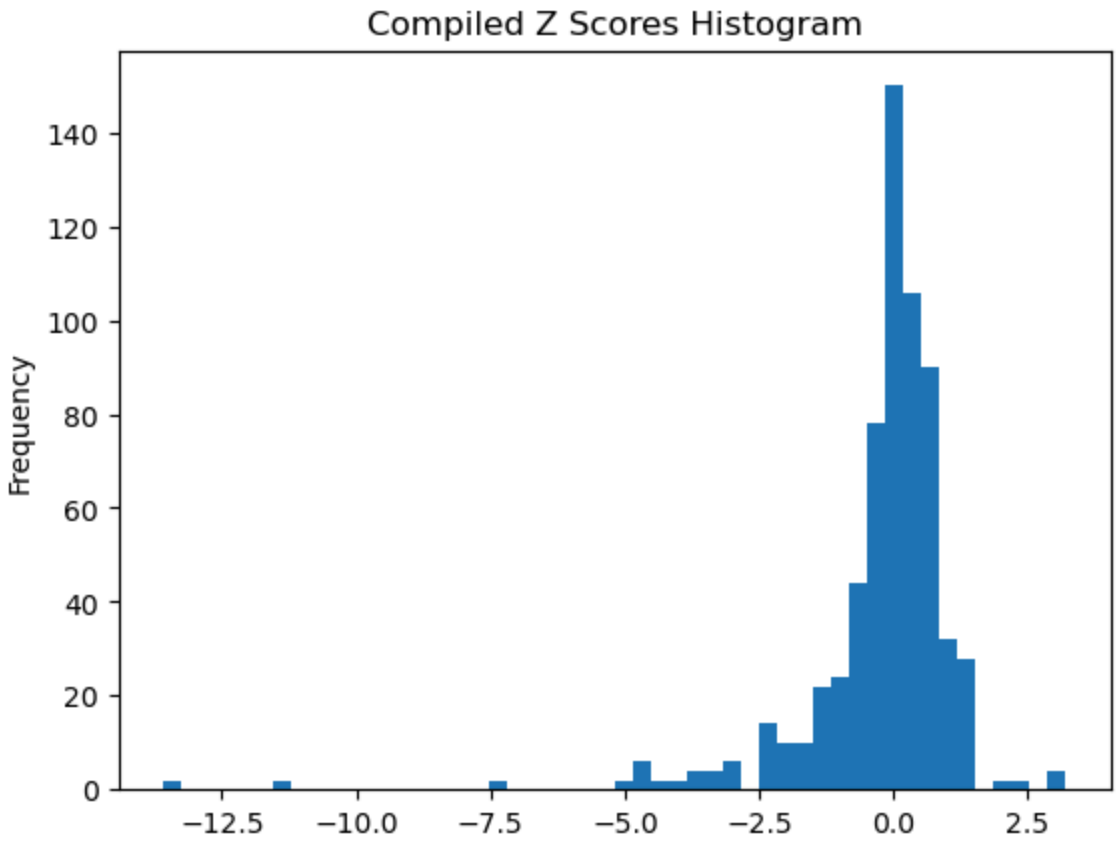
```
range = [-14, 4]
```


S00712
S00724
S00733
S00781
S00794
S00802
S00811
S00877
S00880
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas

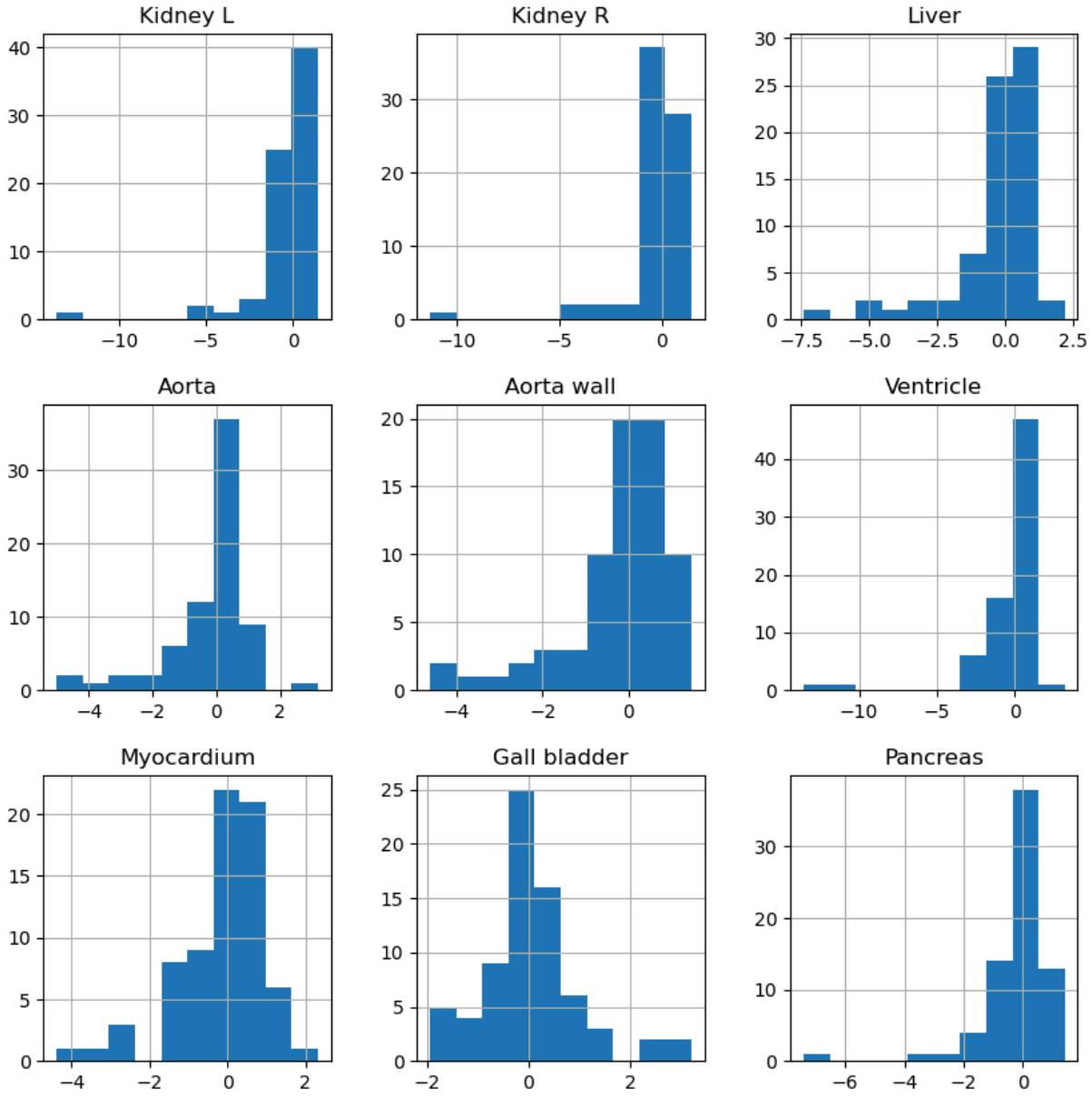
range = [−1.25, 0.5]


Average Z Score for Rifampicin Group, Leave 1 Out Method
Kidney L -0.28 -0.32 -0.36 -0.11 -1.21 -0.19 -0.03 -0.13
Kidney R -0.28 -0.25 -0.33 -0.14 -1.00 -0.22 -0.02 -0.12
Liver -0.32 -0.25 -0.32 -0.30 -0.22 -0.14 0.00 -0.57
Aorta -0.36 -0.33 -0.32 -0.37 -0.18 -0.10 0.09 -0.14
Aorta wall -0.11 -0.14 -0.30 -0.37 -0.28 -0.34 0.06 -0.14
Ventricle -1.21 -1.00 -0.22 -0.18 -0.28 -0.20 0.08 -0.16
Myocardium -0.19 -0.22 -0.14 -0.10 -0.34 -0.20 0.04 -0.21
Gall bladder -0.03 -0.02 0.00 0.09 0.06 0.08 0.04 0.01
Pancreas -0.13 -0.12 -0.57 -0.14 -0.14 -0.16 -0.21 0.01
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
−1.2


Average Z Score Histogram
Frequency
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
−1.2
−1.0
−0.8
−0.6
−0.4
−0.2
0.0



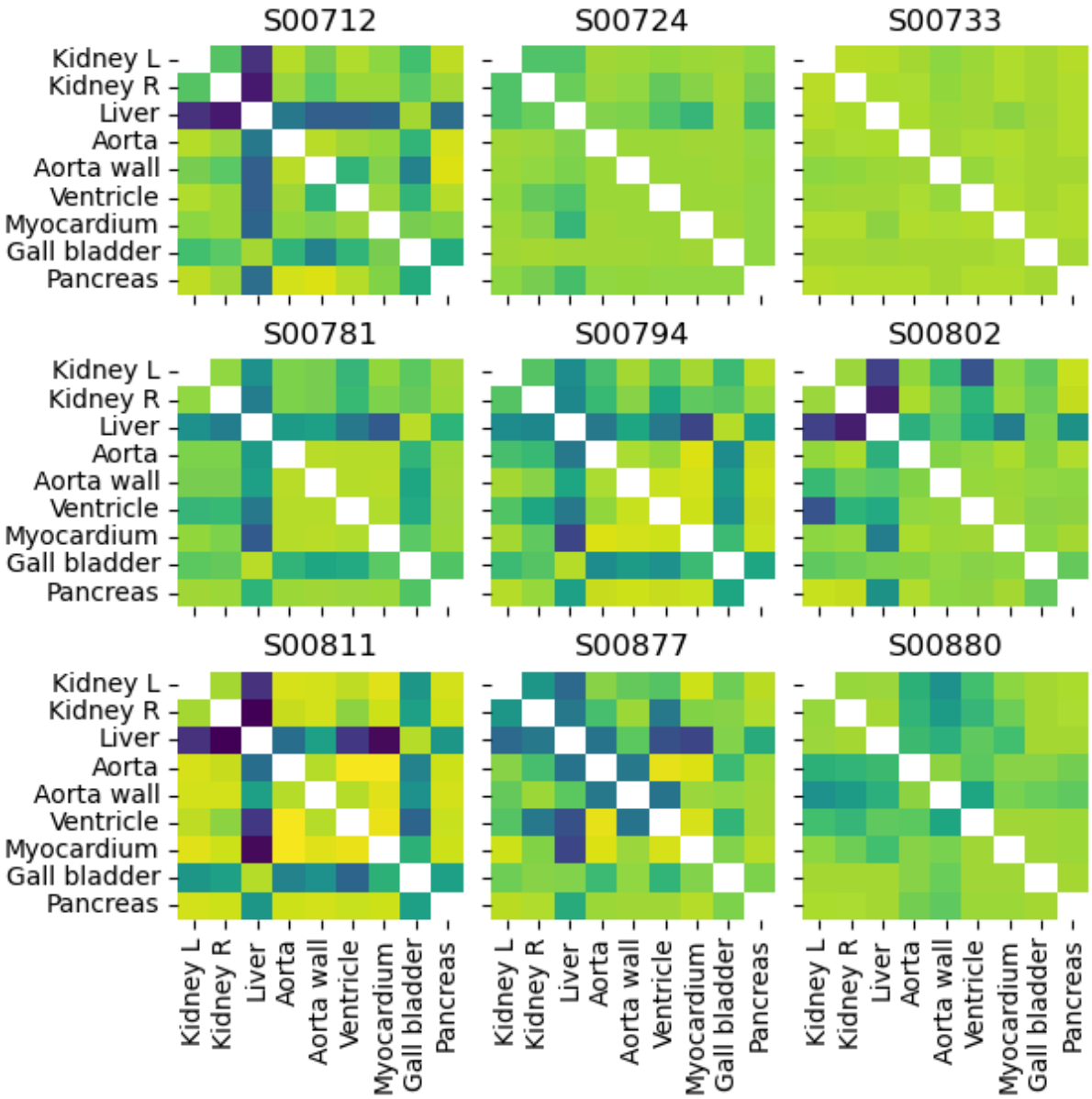
Out[33]:
```
array([[<Axes: title={'center': 'Kidney L'}>,
        <Axes: title={'center': 'Kidney R'}>,
        <Axes: title={'center': 'Liver'}>],
       [<Axes: title={'center': 'Aorta'}>,
        <Axes: title={'center': 'Aorta wall'}>,
        <Axes: title={'center': 'Ventricle'}>],
       [<Axes: title={'center': 'Myocardium'}>,
        <Axes: title={'center': 'Gall bladder'}>,
        <Axes: title={'center': 'Pancreas'}>]], dtype=object)
```



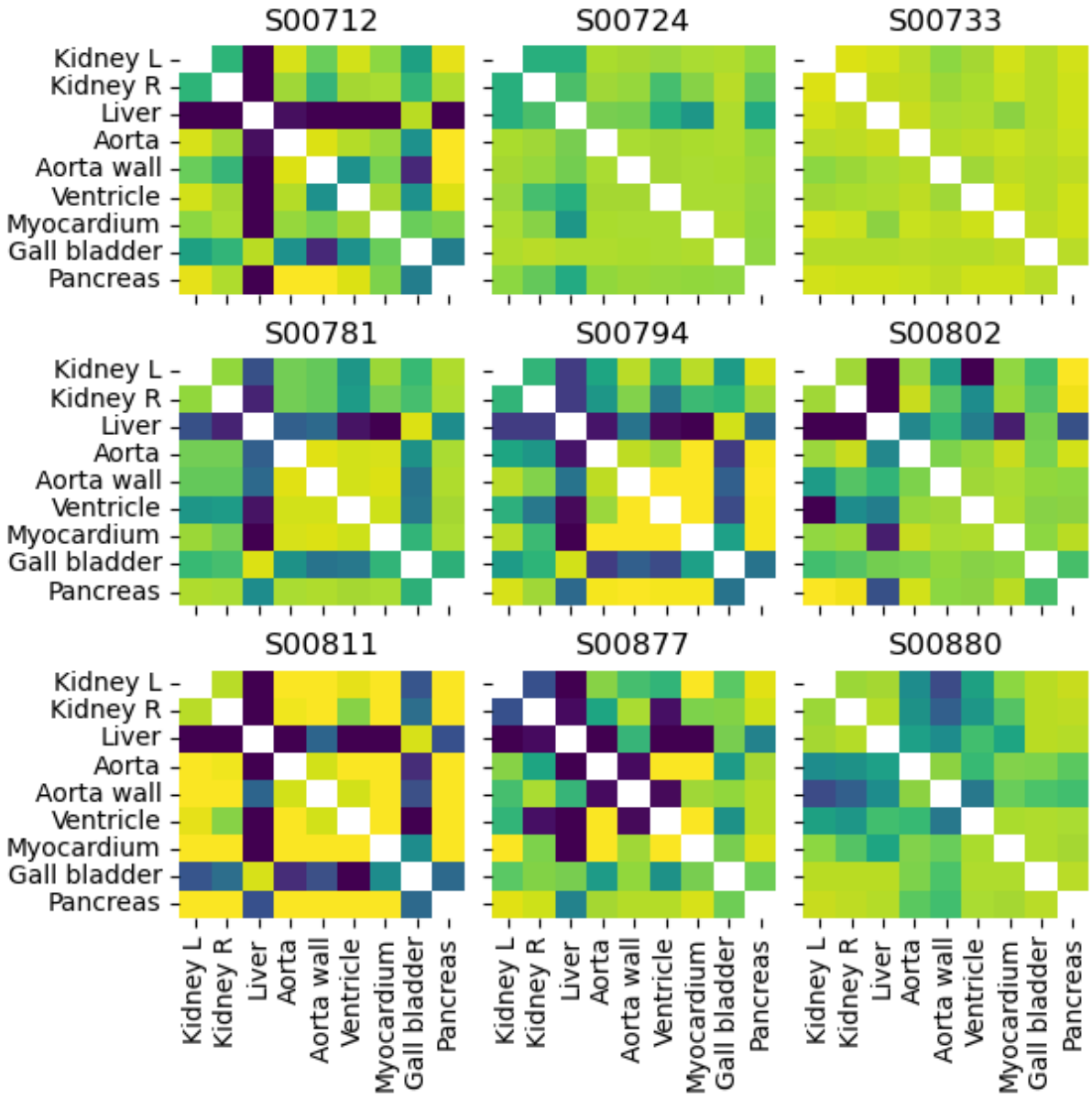
In [37]:
```
#Now the same but the rifampicin subjects into the control group (so, not le
print('Z scores for each rifampicin subject compared to the control group')

ctrl = df.transpose()[ctrlsubs].copy().transpose()
refnet = ctrl.corr(method='pearson')
refnet = refnet.transpose()

fig, axes = plt.subplots(3, 3, sharey=True, sharex=True, figsize=(6.25,6.25)
i=0
zscore_list = []
for subject in rifsubs:
    ptb = df.transpose()[ctrlsubs].copy()
    ptb = ptb.assign(rifsub=df.transpose()[subject])
    ptbnet = ptb.transpose().corr(method='pearson').transpose()
    resnet = ptbnet-refnet
    zscore = resnet/((1-refnet**2)/(len(ctrlsubs)-1))
    # # Method to plot each individual z score map
    # snszscore_indiv = sns.heatmap(zscore, annot=True, vmin=-6, vmax=1, cma
    # plt.title("Z-score compared to controls for "+subject)
```

```
    # plt.show()
    # zscore_list.append(zscore)
    # Method to plot each individual z score map
    snszscore_indiv = sns.heatmap(zscore, vmin=-11, vmax=2, cmap=palette, cb
    axes.flat[i].set_title(subject)
    zscore_list.append(zscore)
    i += 1
fig.tight_layout()
plt.show()
print('range = [-11,2] \n')

#again with different colormap range
fig, axes = plt.subplots(3, 3, sharey=True, sharex=True, figsize=(6.25,6.25)
i=0
zscore_list = []
for subject in rifsubs:
    ptb = df.transpose()[ctrlsubs].copy()
    ptb = ptb.assign(rifsub=df.transpose()[subject])
    ptbnet = ptb.transpose().corr(method='pearson').transpose()
    resnet = ptbnet-refnet
    zscore = resnet/((1-refnet**2)/(len(ctrlsubs)-1))
    # # Method to plot each individual z score map
    # snszscore_indiv = sns.heatmap(zscore, annot=True, vmin=-6, vmax=1, cma
    # plt.title("Z-score compared to controls for "+subject)
    # plt.show()
    # zscore_list.append(zscore)
    # Method to plot each individual z score map
    snszscore_indiv = sns.heatmap(zscore, vmin=-6, vmax=1, cmap=palette, cba
    axes.flat[i].set_title(subject)
    zscore_list.append(zscore)
    i += 1
fig.tight_layout()
plt.show()
print('range = [-6,1] \n')

avgzscore = sum(zscore_list)/len(zscore_list)
# print('avg zscore min = ', avgzscore.stack().min(), ', max = ', avgzscore.

snszscore = sns.heatmap(avgzscore, annot=True, vmin=-6, vmax=1, cmap=palette
plt.title('Average Z Score for Individual Level Analysis of Rif to Ctrls')
plt.show()
avgzscore.stack().plot.hist(bins=10)
plt.title('Average Z Score Histogram')
plt.show()

zcomb = pd.concat(zscore_list, ignore_index=True)
zcomb.stack().plot.hist(bins=50)
plt.title('Compiled Z Scores Histogram')
plt.show()
fig, axis = plt.subplots(3, 3, figsize=(10,10))
zcomb.hist(ax=axis)
```

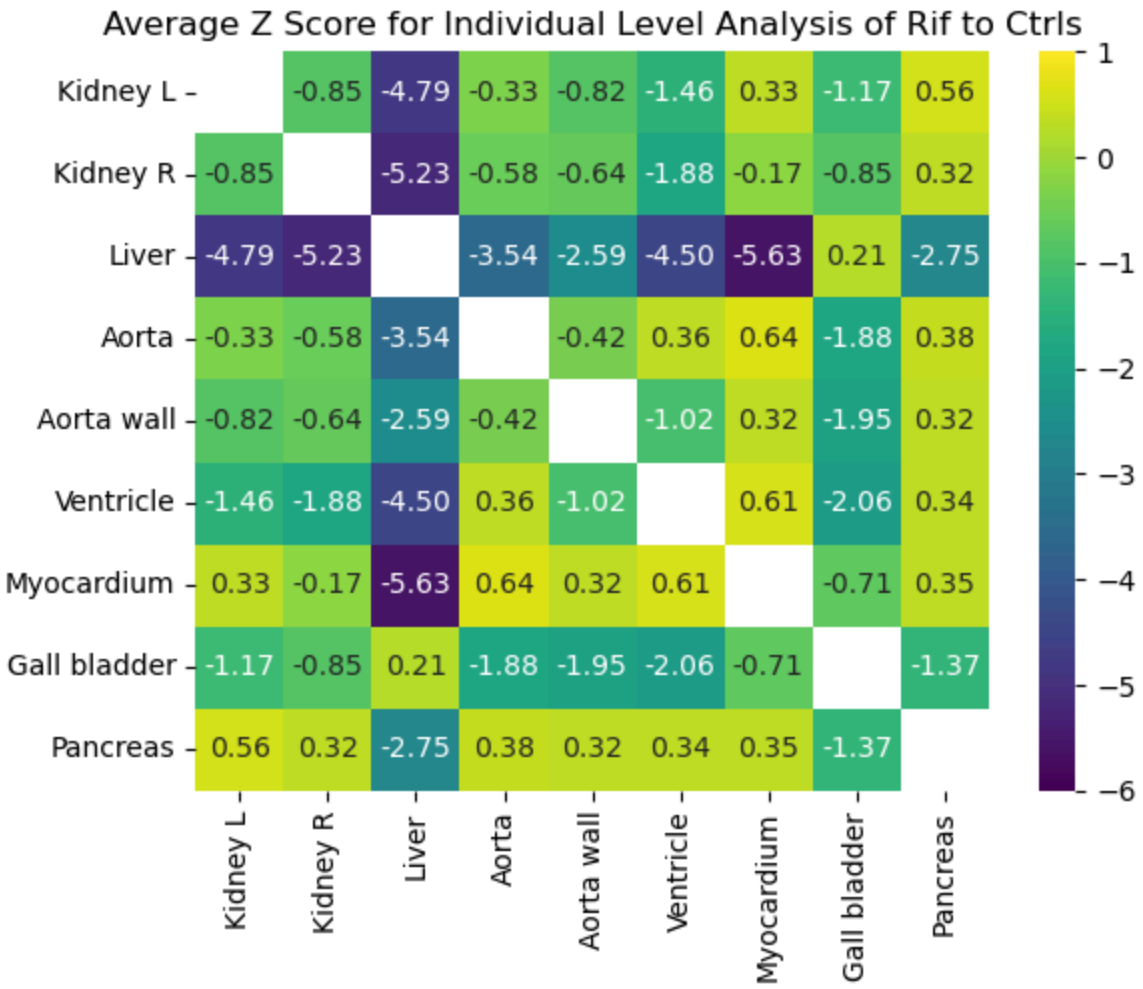
Z scores for each rifampicin subject compared to the control group

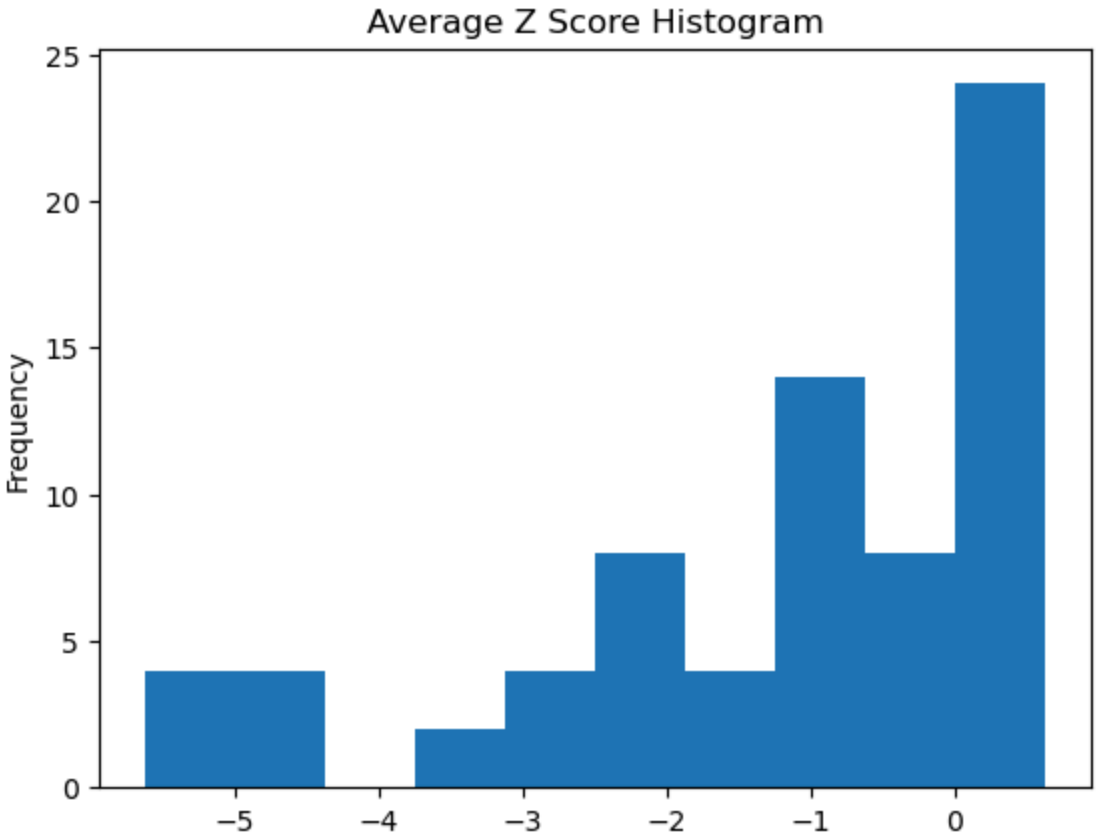

S00712
S00724
S00733
S00781
S00794
S00802
S00811
S00877
S00880
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas

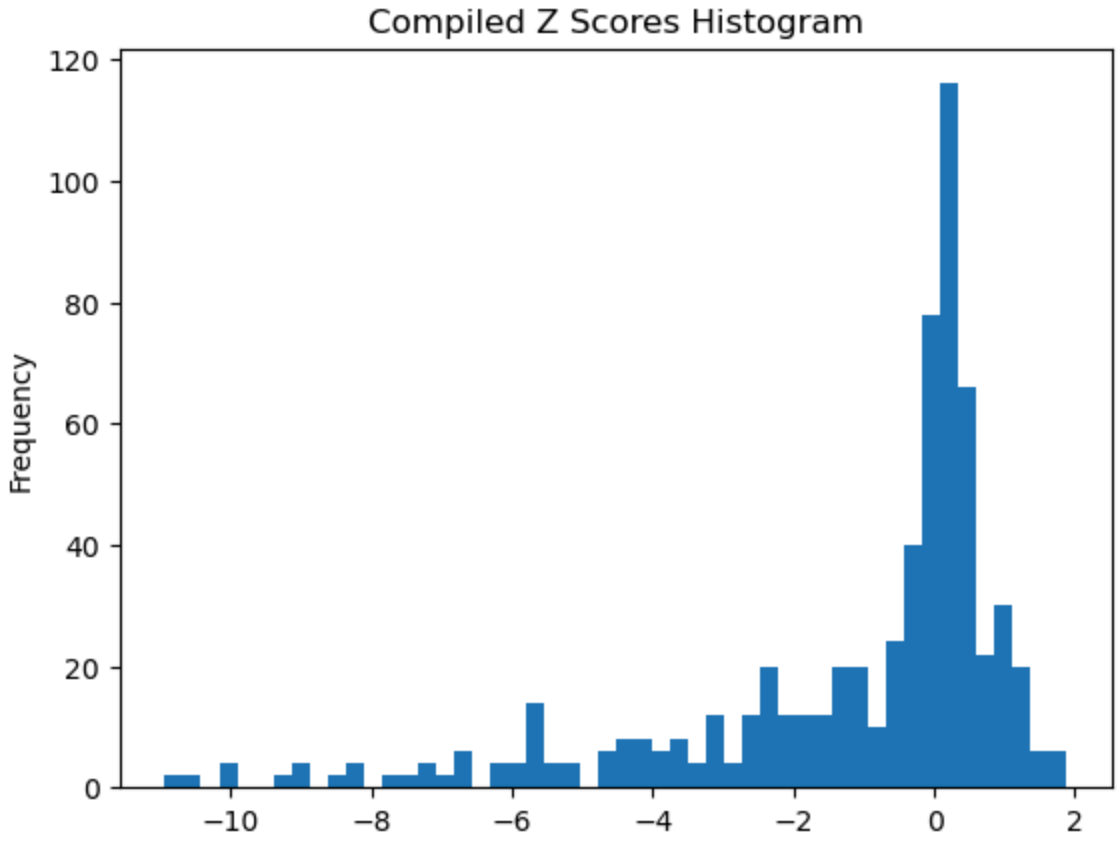
range = [-11,2]


S00712
S00724
S00733
S00781
S00794
S00802
S00811
S00877
S00880
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas

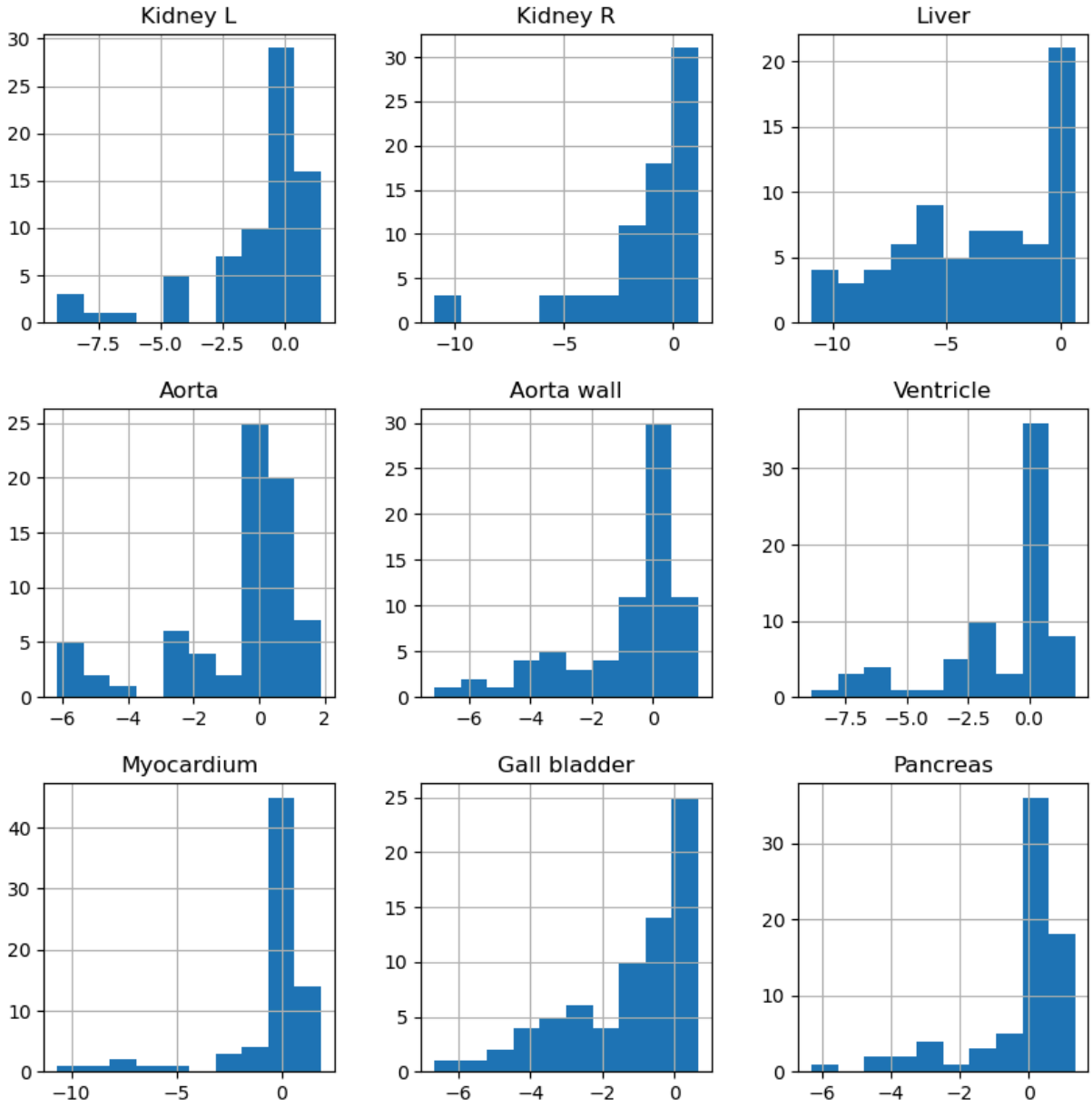
```
range = [-6,1]
```


Average Z Score for Individual Level Analysis of Rif to Ctrls
Kidney L -0.85 -4.79 -0.33 -0.82 -1.46 0.33 -1.17 0.56
Kidney R -0.85 -5.23 -0.58 -0.64 -1.88 -0.17 -0.85 0.32
Liver -4.79 -5.23 -3.54 -2.59 -4.50 -5.63 0.21 -2.75
Aorta -0.33 -0.58 -3.54 -0.42 0.36 0.64 -1.88 0.38
Aorta wall -0.82 -0.64 -2.59 -0.42 -1.02 0.32 -1.95 0.32
Ventricle -1.46 -1.88 -4.50 0.36 -1.02 0.61 -2.06 0.34
Myocardium 0.33 -0.17 -5.63 0.64 0.32 0.61 -0.71 0.35
Gall bladder -1.17 -0.85 0.21 -1.88 -1.95 -2.06 -0.71 -1.37
Pancreas 0.56 0.32 -2.75 0.38 0.32 0.34 0.35 -1.37
Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas
1
0
−1
−2
−3
−4
−5
−6


Average Z Score Histogram
Frequency
25
20
15
10
5
0
−5
−4
−3
−2
−1
0




```
Out[37]: array([[<Axes: title={'center': 'Kidney L'}>,
                 <Axes: title={'center': 'Kidney R'}>,
                 <Axes: title={'center': 'Liver'}>],
                [<Axes: title={'center': 'Aorta'}>,
                 <Axes: title={'center': 'Aorta wall'}>,
                 <Axes: title={'center': 'Ventricle'}>],
                [<Axes: title={'center': 'Myocardium'}>,
                 <Axes: title={'center': 'Gall bladder'}>,
                 <Axes: title={'center': 'Pancreas'}>]], dtype=object)
```


Kidney L
Kidney R
Liver
Aorta
Aorta wall
Ventricle
Myocardium
Gall bladder
Pancreas

In [ ]: