# **ELF\_Cytokine\_Analysis Corr, Apr 23, 2024. Python 3.18**

#--------------------------------------------------------------------------

#ELF\_Cytokine\_Stats Analysis. Corr

# Apr 23, 2024 Python 3.18 by Teiji Sawa, MD, PhD

# Kyoto Prefectural University of Medicine, Japan

# For the analyses of the following publication:

# Authors: Sazuki Sudo, others & Teiji Sawa.

# Title: Case study observational research: inflammatory

# cytokines in the bronchial epithelial lining fluid of COVID-19

# patients with acute hypoxemic respiratory failure.

# Journal: Critical Care volume 28: 134, 2024.

# DOI: doi:10.1186/s13054-024-04921-3

#--------------------------------------------------------------------------

#-**[Python Code #1]**-------------------------------------------------

**import** numpy **as** np

**import** pandas **as** pd

**import** seaborn **as** sns

**import** matplotlib.pyplot **as** plt

#-[END]-----------------------------------------------------------------

#-**[Python Code #2]**-------------------------------------------------

data\_all **=** pd.read\_csv('Data1\_ELF&Plasma\_wNormal\_20240423.csv')

data\_all

#-[END]-----------------------------------------------------------------

**[Out]**

|  | cytokine | group | source | conc |
| --- | --- | --- | --- | --- |
| 0 | IL6 | covid | ELF | 91.28 |
| 1 | IFNg | covid | ELF | 4.39 |
| 2 | MCP1 | covid | ELF | 1538.62 |
| ... | ... | ... | ... | ... |
| 1599 | IL13 | normal | plasma | 0.03 |

1600 rows × 4 columns

#-**[Python Code #3]**-------------------------------------------------

data\_all\_ELF **=** data\_all[data\_all.source**==**'ELF']

data\_all\_ELF

#-[END]-----------------------------------------------------------------

**[Out]**

|  | **cytokine** | **group** | **source** | **conc** |
| --- | --- | --- | --- | --- |
| **0** | IL6 | covid | ELF | 91.280000 |
| **1** | IFNg | covid | ELF | 4.390000 |
| **2** | MCP1 | covid | ELF | 1538.620000 |
| **...** | ... | ... | ... | ... |
| **1574** | IL13 | normal | ELF | 0.030000 |

800 rows × 4 columns

#-**[Python Code #4]**-------------------------------------------------

fig, ax **=** plt.subplots(figsize**=** (12, 10))

sns.boxplot(x**=**'cytokine', y='conc', data**=**data\_all\_ELF, hue**=**'group', \

hue\_order**=** ['normal', 'covid'],

order**=** ['GCSF','GMCSF','IFNg','MCP1','MIP1b','TNFa', \

'sCD40L','IL1b','IL2','IL4','IL5','IL6','IL7','IL8','IL10', \

'IL12p70','IL13','IL17A','IL17F','IL21','IL22','IL23', \

'IL25','IL31','IL33'],

palette**=** ["skyblue", "pink"],

dodge**=True**, color**=**'black', ax**=**ax,

linewidth**=**0.8,

fliersize**=**2

)

sns.stripplot(x**=**'cytokine', y**=**'conc', data**=**data\_all\_ELF, dodge**=True**, \

jitter**=True**, hue**=**'group', hue\_order**=** ['normal', 'covid'],

order**=** ['GCSF','GMCSF','IFNg','MCP1','MIP1b','TNFa', \

'sCD40L','IL1b','IL2','IL4','IL5','IL6','IL7','IL8','IL10', \

'IL12p70','IL13','IL17A','IL17F','IL21','IL22','IL23', \

'IL25','IL31','IL33'],

palette**=** ["blue", "red"], ax**=**ax, size**=**4)

ax.set\_yscale('symlog')

ax.set\_ylim([0, 1000000])

ax.legend()

plt.xticks(rotation**=**90)

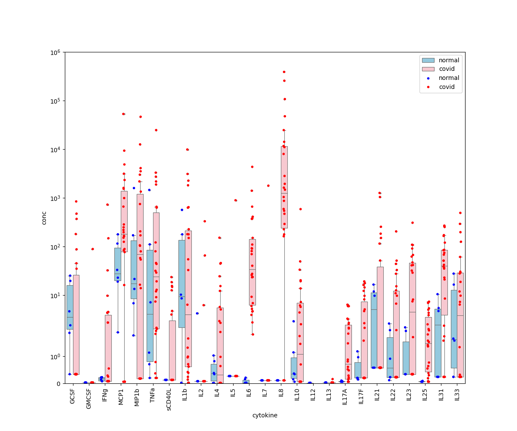
plt.savefig('Fig1\_ELF\_all.svg')

plt.savefig('Fig1\_ELF\_all.png')

plt.show()

#-[END]-----------------------------------------------------------------

**[Out]**



#-**[Python Code #5]**-------------------------------------------------

data\_all\_plasma **=** data\_all[data\_all.source**=**'plasma']

data\_all\_plasma

#-[END]-----------------------------------------------------------------

**[Out]**

|  | cytokine | group | source | conc |
| --- | --- | --- | --- | --- |
| 25 | IL6 | covid | plasma | 17.67 |
| 26 | IFNg | covid | plasma | 0.08 |
| ... | ... | ... | ... | ... |
| 1599 | IL13 | normal | plasma | 0.03 |

800 rows × 4 columns

#-**[Python Code #6]**-------------------------------------------------

fig, ax **=** plt.subplots(figsize=(12, 10))

sns.boxplot(x**=**'cytokine', y**=**'conc', data**=**data\_all\_plasma, \

hue**=**'group', hue\_order**=** ['normal', 'covid'],

order**=** ['GCSF','GMCSF','IFNg','MCP1','MIP1b','TNFa', \

'sCD40L','IL1b','IL2','IL4','IL5','IL6','IL7','IL8','IL10', \

'IL12p70','IL13','IL17A','IL17F','IL21','IL22','IL23', \

'IL25','IL31','IL33'],

palette**=** ["skyblue", "pink"],

dodge**=True**, color**=**'black', ax**=**ax,

linewidth**=**0.8,

fliersize**=**2

)

sns.stripplot(x**=**'cytokine', y**=**'conc', data**=**data\_all\_plasma, dodge**=True**, \

jitter**=True**, hue**=**'group', hue\_order**=** ['normal', 'covid'],

order**=** ['GCSF','GMCSF','IFNg','MCP1','MIP1b','TNFa', \

'sCD40L','IL1b','IL2','IL4','IL5','IL6','IL7','IL8','IL10', \

'IL12p70','IL13','IL17A','IL17F','IL21','IL22','IL23', \

'IL25','IL31','IL33'],

palette**=** ["blue", "red"], ax**=**ax, size**=**4)

ax.set\_yscale('symlog')

ax.set\_ylim([0, 100])

ax.legend()

plt.xticks(rotation**=**90)

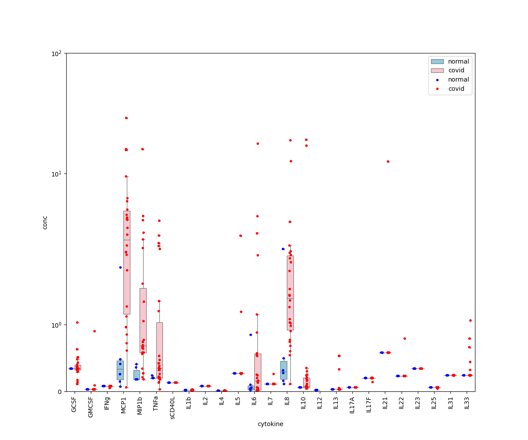
plt.savefig('Fig2\_plasma\_all.svg')

plt.savefig('Fig2\_plasma\_all.png')

plt.show()

#-[END]-----------------------------------------------------------------

**[Out]**



#-**[Python Code #7]**-------------------------------------------------

data\_covid **=** pd.read\_csv('Data2\_ELFperPlasma\_20240423.csv')

data\_covid.head(3)

#-[END]-----------------------------------------------------------------

**[Out]**

|  | cytokine | group | source | elf\_per\_plasma |
| --- | --- | --- | --- | --- |
| 0 | IL6 | covid | ELF | 17.67 |
| 1 | IFNg | covid | ELF | 0.08 |
| 2 | MCP1 | covid | ELF | 15.93 |

#-**[Python Code #8]**-------------------------------------------------

fig, ax **=** plt.subplots(figsize**=** (12, 10))

sns.boxplot(x**=**'cytokine', y**=**'elf\_per\_plasma', data**=**data\_covid,

order**=** ['GCSF','GMCSF','IFNg','MCP1','MIP1b','TNFa', \

'sCD40L','IL1b','IL2','IL4','IL5','IL6','IL7','IL8','IL10', \

'IL12p70','IL13','IL17A','IL17F','IL21','IL22','IL23', \

'IL25','IL31','IL33'],

color**=**'palegreen', ax**=**ax,

linewidth**=**1.2,

fliersize**=**0.1,

)

sns.stripplot(x**=**'cytokine', y**=**'elf\_per\_plasma', data**=**data\_covid, \

jitter**=True**, color**=**'green',

order**=** ['GCSF','GMCSF','IFNg','MCP1','MIP1b','TNFa', \

'sCD40L','IL1b','IL2','IL4','IL5','IL6','IL7','IL8','IL10', \

'IL12p70','IL13','IL17A','IL17F','IL21','IL22','IL23', \

'IL25','IL31','IL33'],

ax**=**ax, size**=**5)

ax.set\_yscale('symlog')

#ax.set\_yscale('log')

ax.set\_ylim([0.01, 1000000])

plt.xticks(rotation**=**90)

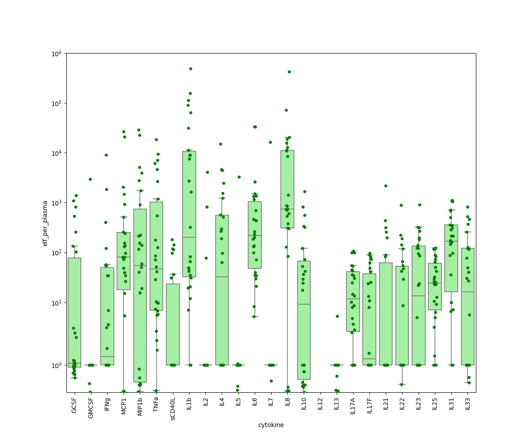
plt.savefig('Fig3\_elf\_per\_plasma.svg')

plt.savefig('Fig3\_elf\_per\_plasma.png')

plt.show()

#-[END]-----------------------------------------------------------------

**[Out]**



# # Correlation Analysis[¶](http://localhost:8978/notebooks/Desktop/Sudo_ELF/data/p27/Cytokine_stats_20240421.ipynb#%E7%9B%B8%E9%96%A2%E8%A7%A3%E6%9E%90)

#-**[Python Code #9]**-------------------------------------------------

data **=** pd.read\_csv('Data3\_ELF\_20240423.csv')

data.head(3)

#-[END]-----------------------------------------------------------------

**[Out]**

| ch\_no | psi\_no | … | PSI | LIV | IL6 |
| --- | --- | --- | --- | --- | --- |
| 0 | 1 | 24 | 1 | 173 | 91.28 |
| 1 | 2 | 27 | 2 | 245 | 371.47 |
| 2 | 3 | 5 | 2 | 75 | 411.25 |

3 rows × 35 columns

#-**[Python Code #10]**-------------------------------------------------

data\_corr **=** data[['PSI','LIV','TNFa','IL1b','IL8','GCF','MIP1b','IL2', \

'IL10','IL4','MCP1','IFNg','IL6','IL33','IL17F','IL22', \

'IL21','IL31','IL23','IL17A','IL25','sCD40L']]

data\_all\_r3 **=** data\_corr.copy()

data\_all\_r3.head(3)

#-[END]-----------------------------------------------------------------

**[Out]**

|  | PSI | …. | IL10 | IL4 | ... |
| --- | --- | --- | --- | --- | --- |
| 0 | 173 | 65.79 | 49.74 | 5.11 |
| 1 | 245 | 0.08 | 33.53 | 0.01 |
| 2 | 75 | 0.08 | 6.20 | 0.01 |

3 rows × 22 columns

#-**[Python Code #11]**-------------------------------------------------

###Logarithmic

# IL6: standardize

X1 **=** data\_all\_r3.loc[:, ['IL6']]

X1 **=** X1.apply(np.log)

# IL8: standardize

X2 **=** data\_all\_r3.loc[:, ['IL8']]

X2 **=** X2.apply(np.log)

# MCP1: standardize

X3 **=** data\_all\_r3.loc[:, ['MCP1']]

X3 **=** X3.apply(np.log)

# MIP1b: standardize

X4 **=** data\_all\_r3.loc[:, ['MIP1b']]

X4 **=** X4.apply(np.log)

# TNFa: standardize

X5 **=** data\_all\_r3.loc[:, ['TNFa']]

X5 **=** X5.apply(np.log)

# IL10: standardize

X6 **=** data\_all\_r3.loc[:, ['IL10']]

X6 **=** X6.apply(np.log)

# IL33: standardize

X7 **=** data\_all\_r3.loc[:, ['IL33']]

X7 **=** X7.apply(np.log)

# IFNg: standardize

X8 **=** data\_all\_r3.loc[:, ['IFNg']]

X8 **=** X8.apply(np.log)

# IL1b: standardize

X9 **=** data\_all\_r3.loc[:, ['IL1b']]

X9 **=** X9.apply(np.log)

# GCF: standardize

X10 **=** data\_all\_r3.loc[:, ['GCF']]

X10 **=** X10.apply(np.log)

# IL4: standardize

X11 **=** data\_all\_r3.loc[:, ['IL4']]

X11 **=** X11.apply(np.log)

# IL17F: standardize

X12 **=** data\_all\_r3.loc[:, ['IL17F']]

X12 **=** X12.apply(np.log)

# IL21: standardize

X13 **=** data\_all\_r3.loc[:, ['IL21']]

X13 **=** X13.apply(np.log)

# IL22: standardize

X14 **=** data\_all\_r3.loc[:, ['IL22']]

X14 **=** X14.apply(np.log)

# IL17A: standardize

X15 **=** data\_all\_r3.loc[:, ['IL17A']]

X15 **=** X15.apply(np.log)

# IL23: standardize

X16 **=** data\_all\_r3.loc[:, ['IL23']]

X16 **=** X16.apply(np.log)

# IL25: standardize

X17 = data\_all\_r3.loc[:, ['IL25']]

X17 = X17.apply(np.log)

# IL31: standardize

X18 **=** data\_all\_r3.loc[:, ['IL31']]

X18 **=** X18.apply(np.log)

# IL2: standardize

X19 **=** data\_all\_r3.loc[:, ['IL2']]

X19 **=** X19.apply(np.log)

# sCD40L: standardize

X24 **=** data\_all\_r3.loc[:, ['sCD40L']]

X24 **=** X24.apply(np.log)

# Output

data\_all\_r3.loc[:, ['IL6']] **=** X1

data\_all\_r3.loc[:, ['IL8']] **=** X2

data\_all\_r3.loc[:, ['MCP1']] **=** X3

data\_all\_r3.loc[:, ['MIP1b']] **=** X4

data\_all\_r3.loc[:, ['TNFa']] **=** X5

data\_all\_r3.loc[:, ['IL10']] **=** X6

data\_all\_r3.loc[:, ['IL33']] **=** X7

data\_all\_r3.loc[:, ['IFNg']] **=** X8

data\_all\_r3.loc[:, ['IL1b']] **=** X9

data\_all\_r3.loc[:, ['GCF']] **=** X10

data\_all\_r3.loc[:, ['IL4']] **=** X11

data\_all\_r3.loc[:, ['IL17F']] **=** X12

data\_all\_r3.loc[:, ['IL21']] **=** X13

data\_all\_r3.loc[:, ['IL22']] **=** X14

data\_all\_r3.loc[:, ['IL17A']] **=** X15

data\_all\_r3.loc[:, ['IL23']] **=** X16

data\_all\_r3.loc[:, ['IL25']] **=** X17

data\_all\_r3.loc[:, ['IL31']] **=** X18

data\_all\_r3.loc[:, ['IL2']] **=** X19

data\_all\_r3.loc[:, ['sCD40L']] **=** X24

#-[END]-----------------------------------------------------------------

#-**[Python Code #12]**-------------------------------------------------

data\_all\_r3.head(3)

#-[END]-----------------------------------------------------------------

**[Out]**

|  | PSI | … | IL10 |
| --- | --- | --- | --- |
| 0 | 173 | 4.186468 | 3.906809 |
| 1 | 245 | -2.525729 | 3.512441 |
| 2 | 75 | -2.525729 | 1.824549 |

3 rows × 22 columns

#-**[Python Code #13]**-------------------------------------------------

data\_all\_r3 **=** data\_all\_r3.iloc[:,:]

corr **=** data\_all\_r3.corr()

print(corr)

#-[END]-----------------------------------------------------------------

**[Out]**

PSI LIV TNFa IL1b IL8 GCF MIP1b \

PSI 1.000000 0.346204 0.018715 0.166074 0.207175 0.030441 -0.011050

LIV 0.346204 1.000000 -0.484264 -0.325923 -0.278428 -0.175616 -0.453120

TNFa 0.018715 -0.484264 1.000000 0.830292 0.629645 0.570032 0.701376

…..

sCD40L -0.229752 -0.408536 0.446174 0.444807 0.156168 0.027222 0.218299

IL2 IL10 IL4 ... IL6 IL33 IL17F \

PSI -0.108708 -0.217492 -0.162748 ... -0.085738 -0.189133 0.049929

LIV -0.196742 -0.392568 -0.259178 ... -0.237532 -0.437492 -0.154480

TNFa 0.572178 0.731840 0.606350 ... 0.640329 0.263769 0.447039

…..

sCD40L 0.205723 0.223809 0.459109 ... 0.182824 0.196474 0.248289

IL22 IL21 IL31 IL23 IL17A IL25 sCD40L

PSI -0.077068 -0.119359 -0.195992 -0.131076 -0.109583 -0.100648 -0.229752

LIV -0.247969 -0.153879 -0.037292 -0.156624 0.062991 0.047808 -0.408536

TNFa 0.443901 0.335365 0.275557 0.270827 0.149964 0.250751 0.446174

…..

sCD40L 0.385023 0.160526 0.519911 0.646726 0.573544 0.518472 1.000000

[22 rows x 22 columns]

#-**[Python Code #14]**-------------------------------------------------

def display\_correlation(df):

r **=** df.corr(method="spearman")

plt.figure(figsize**=** (12,10))

heatmap **=** sns.heatmap(df.corr(), vmin**=**-1,

vmax**=**1, annot**=True**, cmap**=**'jet', \

fmt**=**'.2f', annot\_kws**=**{'fontsize': 12},)

plt.title("Spearman Correlation")

return(r)

#-[END]-----------------------------------------------------------------

#-**[Python Code #15]**-------------------------------------------------

sns.set(font\_scale**=**1.2)

display\_correlation(corr)

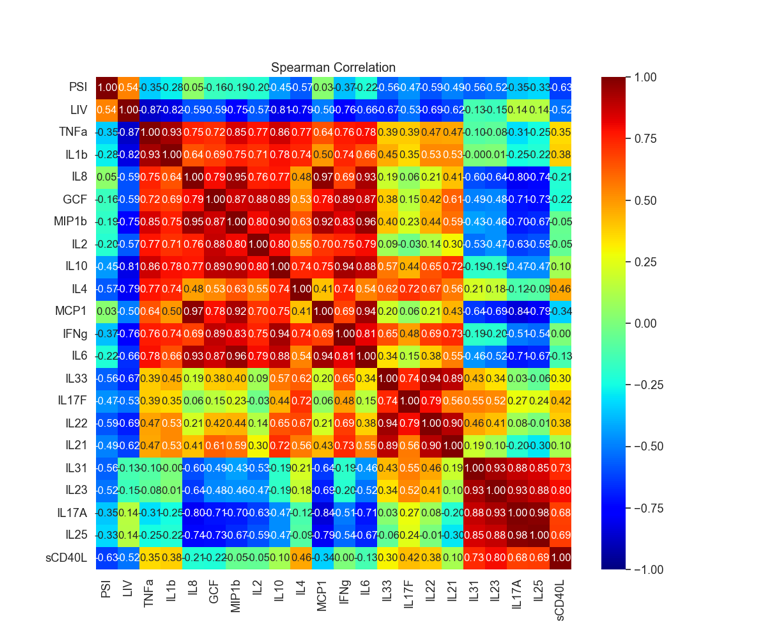
plt.savefig('Fig4\_spearman\_elf.svg')

plt.savefig('Fig4\_spearman\_elf.png')

plt.show()

#-[END]-----------------------------------------------------------------

**[Out]**



#-**[Python Code #16]**-------------------------------------------------

data2 = pd.read\_csv('Data4\_Plasma\_20240423.csv')

data2.head(3)

#-[END]-----------------------------------------------------------------

**[Out]**

|  | ch\_no | ….. | PSI |
| --- | --- | --- | --- |
| 0 | 1 | 1 | 173 |
| 1 | 2 | 2 | 245 |
| 2 | 3 | 2 | 75 |

3 rows × 35 columns

#-**[Python Code #17]**-------------------------------------------------

data2\_corr **=** data2[['PSI','LIV','IL8','MCP1','IL6','TNFa','IL13','IL33', \

'MIP1b','IL5','GMCF','GCF','IL10']]

data2\_all\_r3 **=** data2\_corr.copy()

#-[END]-----------------------------------------------------------------

#-**[Python Code #18]**-------------------------------------------------

###Logarithmic

# IL6: standardize

X1 **=** data2\_all\_r3.loc[:, ['IL6']]

X1 **=** X1.apply(np.log)

# IL8: standardize

X2 **=** data2\_all\_r3.loc[:, ['IL8']]

X2 **=** X2.apply(np.log)

# MCP1: standardize

X3 **=** data2\_all\_r3.loc[:, ['MCP1']]

X3 **=** X3.apply(np.log)

# MIP1b: standardize

X4 **=** data2\_all\_r3.loc[:, ['MIP1b']]

X4 **=** X4.apply(np.log)

# TNFa: standardize

X5 **=** data2\_all\_r3.loc[:, ['TNFa']]

X5 **=** X5.apply(np.log)

# IL10: standardize

X6 **=** data2\_all\_r3.loc[:, ['IL10']]

X6 **=** X6.apply(np.log)

# IL33: standardize

X7 **=** data2\_all\_r3.loc[:, ['IL33']]

X7 **=** X7.apply(np.log)

# GCF: standardize

X10 **=** data2\_all\_r3.loc[:, ['GCF']]

X10 **=** X10.apply(np.log)

## GMCF: standardize

X25 **=** data2\_all\_r3.loc[:, ['GMCF']]

X25 **=** X25.apply(np.log)

# Output

data2\_all\_r3.loc[:, ['IL6']] = X1

data2\_all\_r3.loc[:, ['IL8']] = X2

data2\_all\_r3.loc[:, ['MCP1']] = X3

data2\_all\_r3.loc[:, ['MIP1b']] = X4

data2\_all\_r3.loc[:, ['TNFa']] = X5

data2\_all\_r3.loc[:, ['IL10']] **=** X6

data2\_all\_r3.loc[:, ['IL33']] **=** X7

data2\_all\_r3.loc[:, ['GCF']] **=** X10

data2\_all\_r3.loc[:, ['GMCF']] **=** X25

#-[END]-----------------------------------------------------------------

#-**[Python Code #19]**-------------------------------------------------

corr2 **=** data2\_all\_r3.corr()

print(corr2)

#-[END]-----------------------------------------------------------------

**[Out]**

PSI LIV IL8 MCP1 IL6 TNFa IL13 \

PSI 1.000000 0.346204 0.488543 0.494825 0.142918 0.367449 0.341706

LIV 0.346204 1.000000 -0.027189 -0.140894 -0.143893 -0.124760 0.481838

IL8 0.488543 -0.027189 1.000000 0.729544 0.468561 0.612345 -0.062737

….

IL10 -0.281407 0.186937 -0.022785 0.066774 0.050246 -0.135643 -0.095920

IL33 MIP1b IL5 GMCF GCF IL10

PSI 0.096084 0.140068 0.006554 0.002069 -0.078244 -0.281407

LIV -0.144173 -0.033820 0.188067 0.183444 -0.264280 0.186937

IL8 0.426468 0.482982 0.175723 0.175037 0.098238 -0.022785

….

IL10 -0.182520 0.575821 -0.027637 -0.028236 0.104043 1.000000

#-**[Python Code #20]**-------------------------------------------------

sns.set(font\_scale **=**1.2)

display\_correlation(corr2)

plt.savefig('Fig5\_plasma\_spearman.svg')

plt.savefig('Fig5\_plasma\_spearman.png')

plt.show()

#-[END]-----------------------------------------------------------------

**[Out]**

