Target heatmap

[

heatmap

1920×1920 493 KB

](https://forum.numer.ai/uploads/default/original/2X/1/14f1ae32938b185abf257e209e2ae77a4e1d84c6.jpeg)

Target clustermap

[

cluster

3590×3590 302 KB

](https://forum.numer.ai/uploads/default/original/2X/8/832c3b2fa4cf3524f55749f071a20a72a9994bd9.png)

import pandas as pd import seaborn as sns import matplotlib.pyplot as plt

targets = ['target_nomi_v4_20', 'target_nomi_v4_60', 'target_tyler_v4_20', 'target_tyler_v4_60', 'target_victor_v4_20', 'target_victor_v4_60', 'target_ralph_v4_20', 'target_ralph_v4_60', 'target_waldo_v4_20', 'target_waldo_v4_60', 'target_jerome_v4_20', 'target_jerome_v4_60', 'target_janet_v4_20', 'target_janet_v4_60', 'target_ben_v4_20', 'target_ben_v4_60', 'target_alan_v4_20', 'target_alan_v4_60', 'target_paul_v4_20', 'target_paul_v4_60', 'target_george_v4_20', 'target_george_v4_60', 'target_william_v4_20', 'target_william_v4_60', 'target_arthur_v4_20', 'target_arthur_v4_60', 'target_thomas_v4_20', 'target_thomas_v4_60', 'target_cyrus_v4_20', 'target_cyrus_v4_60', 'target_caroline_v4_20', 'target_caroline_v4_60', 'target_sam_v4_20', 'target_sam_v4_60', 'target_xerxes_v4_20', 'target_xerxes_v4_60', 'target_alpha_v4_20', 'target_alpha_v4_60', 'target_bravo_v4_20', 'target_bravo_v4_60', 'target_charlie_v4_20', 'target_charlie_v4_60', 'target_delta_v4_20', 'target_delta_v4_60', 'target_echo_v4_20', 'target_echo_v4_60', 'target_jeremy_v4_20', 'target_jeremy_v4_60', 'target_teager_v4_20', 'target_teager_v4_60', 'target_agnes_v4_20', 'target_agnes_v4_60', 'target_claudia_v4_20', 'target_claudia_v4_60', 'target_rowan_v4_20', 'target_rowan_v4_60']

analyse the validation data, but we could do the same on the training data

df = pd.read_parquet('v4.2/validation_int8.parquet', columns=targets + ['era'])

compute the mean of the era correlation of every target with any other target

corr = df.groupby('era').corr(method='spearman').mean(axis=0, level=1)

arrange the order of the columns and rows (for visualization) so that they

are sorted by correlation with the target 'target_cyrus_v4_20'

corr = corr.sort_values('target_cyrus_v4_20', axis=0, ascending=False).sort_values('target_cyrus_v4_20', axis=1, ascending=False)

plt.rcParams["figure.figsize"] = [24,24] # default is [6.4, 4.8]

ax = sns.heatmap(corr, annot=True) ax.get_figure().savefig('heatmap.png')

sns.clustermap(corr, figsize=(36,36)).savefig('cluster.png')