Code

July 13, 2021

In [1]:
```
import pandas as pd
import scipy.sparse as sprs
import vk
import time
from tqdm.auto import tqdm
import os
import gc
import numpy as np
from scipy.sparse import csr_matrix, lil_matrix
import matplotlib.pyplot as plt
import networkx as nx
import seaborn as sns
import scipy
import random
from scipy import stats
import plotly.graph_objs as go
from scipy.stats import binom
import math
from matplotlib import mlab
import pylab
from statsmodels.stats.proportion import proportion_confint
from sklearn.preprocessing import normalize
from statsmodels.stats.proportion import proportion_confint

from sklearn import datasets, linear_model
from sklearn.metrics import mean_squared_error, r2_score
from tabulate import tabulate

def get_fr(adj_matrix, user_id):
    return np.array(adj_matrix.indices[adj_matrix.indptr[user_id]:adj_matrix.indptr[us

# Data Download
def choose_time_step(data, t1, t2):

    """
    return vectors of numerical and logical variables that correspond to the time step
    """
```

```
    ths_op = np.arange(0, 1.2, 0.2)

    ########################################################

    op1 = f'ops{t1}'
    op2 = f'ops{t2}'
    fr_op1 = f'fr_ops{t1}_avg'
    fr_op2 = f'fr_ops{t2}_avg'
    fr_op1_divergence = f'fr_ops{t1}_std'

    ########################################################

    op_change = data[op2] - data[op1]
    anchor = data[fr_op1] - data[op1]
    mask_move = abs(op_change) > 0.05
    mask_neg = op_change * anchor < 0
    mask_pos = op_change * anchor > 0
    mask_skip = mask_pos & (abs(op_change) > abs(anchor))
    mask_no_skip = mask_pos & (abs(op_change) <= abs(anchor))
    mask_control_fr = abs(data[fr_op1] - data[fr_op2]) < 0.05

    ########################################################

    masks_now = []
    masks_future = []
    masks_fr = []

    #for i in tqdm(range(len(ths_op) - 1)):
    for i in range(len(ths_op) - 1):

        if i < len(ths_op)  - 2:
            mask_temp_curr_pos = (ths_op[i] <= data[op1]) & (data[op1] < ths_op[i+1])
            mask_temp_fut_pos = (ths_op[i] <= data[op2]) & (data[op2] < ths_op[i+1])
            mask_temp_fr_pos = (ths_op[i] <= data[fr_op1]) & (data[fr_op1] < ths_op[i+
        else:
            mask_temp_curr_pos = (ths_op[i] <= data[op1]) & (data[op1] <= ths_op[i+1])
            mask_temp_fut_pos = (ths_op[i] <= data[op2]) & (data[op2] <= ths_op[i+1])
            mask_temp_fr_pos = (ths_op[i] <= data[fr_op1]) & (data[fr_op1] <= ths_op[i

        masks_now.append(mask_temp_curr_pos)
        masks_future.append(mask_temp_fut_pos)
        masks_fr.append(mask_temp_fr_pos)

    return op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, m



def fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom):
```

```
    res = np.zeros((len(ths_op)-1, len(ths_fr_op)-1))
    res_aux_l = np.zeros((len(ths_op)-1, len(ths_fr_op)-1))
    res_aux_r = np.zeros((len(ths_op)-1, len(ths_fr_op)-1))

    for i in range(len(ths_op) - 1):
        for j in range(len(ths_fr_op) - 1):

            mask_slf_pos = (ths_op[i] < data[op1]) & (data[op1] <= ths_op[i+1])
            mask_fr_avg_op = (ths_fr_op[j] < data[fr_op1]) & (data[fr_op1] <= ths_fr_op

            nom = sum(mask_slf_pos & mask_fr_avg_op & mask_nom)
            denom = sum(mask_slf_pos & mask_fr_avg_op & mask_denom)

            if denom>0:
                p = nom / denom
                (conf_l, conf_r) = proportion_confint(nom, denom, alpha=0.05, method='w
            else:
                p = None
                (conf_l, conf_r) = (None, None)

            res[i, j] = p
            res_aux_l[i, j] = conf_l
            res_aux_r[i, j] = conf_r

    return res, res_aux_l, res_aux_r


def aux_plot_EPOC(title_value, k, subplot, x, ths_op, tables1, tables2):

    subplot.set_title(title_value, fontsize=30)

    subplot.plot(x, tables1[0][k], color='blue', marker='*', label='$EPOC_{-}$',
                 linewidth=3, markersize=12)
    subplot.fill_between(x, tables1[1][k], tables1[2][k], color='royalblue', alpha=.1)


    subplot.plot(x, tables2[0][k], color='k', marker='o', label='$EPOC_{+}$',
                 linewidth=3, markersize=12)
    subplot.fill_between(x, tables2[1][k], tables2[2][k], color='dimgray', alpha=.1)

    subplot.axvline(x=ths_op[k], color='k', linestyle='--')
    subplot.axvline(x=ths_op[k+1], color='k', linestyle='--')
    subplot.axvspan(ths_op[k], ths_op[k+1], facecolor='lightcoral', alpha=0.5)

def aux_plot_EPOC_plus_compare_SL_M(subplot, x, tables):

    subplot.plot(x, tables[0][0], color='k', marker='o', label='$x_{i}=SL$',
```

```
                linewidth=3, markersize=12)
    subplot.fill_between(x, tables[1][0], tables[2][0], color='dimgray', alpha=.1)

    subplot.plot(x, tables[0][1], color='b', marker='*', label='$x_{i}=L$',
                linewidth=3, markersize=12)
    subplot.fill_between(x, tables[1][1], tables[2][1], color='royalblue', alpha=.1)

    subplot.plot(x, tables[0][2], color='darkgreen', marker='>', label='$x_{i}=M$',
                linewidth=3, markersize=12)
    subplot.fill_between(x, tables[1][2], tables[2][2], color='green', alpha=.1)

    subplot.grid()


def aux_plot_EPOC_plus_compare_M_SC(subplot, x, tables):

    subplot.plot(x, tables[0][2], color='darkgreen', marker='>', label='$x_{i}=M$',
                linewidth=3, markersize=12)
    subplot.fill_between(x, tables[1][2], tables[2][2], color='green', alpha=.1)

    subplot.plot(x, tables[0][3], color='darkorange', marker='*', label='$x_{i}=C$',
                linewidth=3, markersize=12)
    subplot.fill_between(x, tables[1][3], tables[2][3], color='orange', alpha=.1)

    subplot.plot(x, tables[0][4], color='r', marker='o', label='$x_{i}=SC$',
                linewidth=3, markersize=12)
    subplot.fill_between(x, tables[1][4], tables[2][4], color='coral', alpha=.1)

    subplot.grid()


def aux_plot_EPOC_control_neg(title_value, k, subplot, x, ths_op, tables1, tables2):

    subplot.set_title(title_value, fontsize=30)

    subplot.plot(x, tables1[0][k], color='blue', marker='*', #label='$EPOC_{-}$',
                linewidth=3, markersize=12)
    subplot.fill_between(x, tables1[1][k], tables1[2][k], color='royalblue', alpha=.1)


    subplot.plot(x, tables2[0][k], color='purple', marker='o', #label='$EPOC_{+}$',
                linewidth=3, markersize=12)
    subplot.fill_between(x, tables2[1][k], tables2[2][k], color='darkviolet', alpha=.1

    min_value = min(tables1[0][k].min(), tables2[0][k].min()) - 0.05
    max_value = min(tables1[0][k].max(), tables2[0][k].max()) + 0.05

    subplot.set_ylim([min_value/2, max_value*2])
```

```
    subplot.axvline(x=ths_op[k], color='k', linestyle='--')
    subplot.axvline(x=ths_op[k+1], color='k', linestyle='--')
    subplot.axvspan(ths_op[k], ths_op[k+1], facecolor='lightcoral', alpha=0.5)

def aux_plot_EPOC_control_pos(title_value, k, subplot, x, ths_op, tables1, tables2):

    subplot.set_title(title_value, fontsize=30)

    subplot.plot(x, tables1[0][k], color='k', marker='*', #label='$EPOC_{-}$',
                 linewidth=3, markersize=12)
    subplot.fill_between(x, tables1[1][k], tables1[2][k], color='dimgray', alpha=.1)


    subplot.plot(x, tables2[0][k], color='darkgreen', marker='o', #label='$EPOC_{+}$',
                 linewidth=3, markersize=12)
    subplot.fill_between(x, tables2[1][k], tables2[2][k], color='green', alpha=.1)

    min_value = min(tables1[0][k].min(), tables2[0][k].min()) - 0.05
    max_value = min(tables1[0][k].max(), tables2[0][k].max()) + 0.05

    subplot.set_ylim([min_value/2, max_value*2])

    subplot.axvline(x=ths_op[k], color='k', linestyle='--')
    subplot.axvline(x=ths_op[k+1], color='k', linestyle='--')
    subplot.axvspan(ths_op[k], ths_op[k+1], facecolor='lightcoral', alpha=0.5)



def cross_group_movs_probs(masks_now, masks_future, masks_fr, control_mask, mask_move):

    res = []
    res_l = []
    res_r = []

    for i in tqdm(range(len(masks_now))):

        new_res = np.zeros((len(masks_future), len(masks_fr)))
        new_res_aux_l = np.zeros((len(masks_future), len(masks_fr)))
        new_res_aux_r = np.zeros((len(masks_future), len(masks_fr)))

        for j in range(len(masks_future)):
            for k in range(len(masks_fr)):

                if i != j:
                    nom = sum(masks_now[i] & masks_future[j] & masks_fr[k] & control_ma
                if i == j:
                    nom = sum(masks_now[i] & masks_future[j] & masks_fr[k] & control_ma
```

```
                denom = sum(masks_now[i] & masks_fr[k] & control_mask)

                if denom>0:
                    p = nom / denom
                    (conf_l, conf_r) = proportion_confint(nom, denom, alpha=0.05, metho
                else:
                    p = None
                    (conf_l, conf_r) = (None, None)

                new_res[j, k] = p
                new_res_aux_l[j, k] = conf_l
                new_res_aux_r[j, k] = conf_r

        res.append(new_res)
        res_l.append(new_res_aux_l)
        res_r.append(new_res_aux_r)

    return res, res_l, res_r

def aux_plot(l, m, x, res, res_l, res_r, i, j,
             color_value, fill_color_value,
             marker_value,
             label_value, names):


    if j > i+1:

        ax[l, m].axvline(x=ths_op[0], color='k', linestyle='--')
        ax[l, m].axvline(x=ths_op[i], color='k', linestyle='--')
        ax[l, m].axvline(x=ths_op[i+1], color='k', linestyle='--')
        ax[l, m].axvline(x=ths_op[j], color='k', linestyle='--')
        ax[l, m].axvline(x=ths_op[-1], color='k', linestyle='--')

        if i>0:
            ax[l, m].axvspan(ths_op[0], ths_op[i], facecolor='darkgreen', alpha=0.5)

        ax[l, m].axvspan(ths_op[i+1], ths_op[j], facecolor='lightgreen', alpha=0.5)
        ax[l, m].axvspan(ths_op[j], ths_op[-1], facecolor='lime', alpha=0.5)

    if (j > i) & ~(j > i+1):

        ax[l, m].axvline(x=ths_op[0], color='k', linestyle='--')
        ax[l, m].axvline(x=ths_op[i], color='k', linestyle='--')
        ax[l, m].axvline(x=ths_op[i+1], color='k', linestyle='--')
        #ax[l, m].axvline(x=ths_op[j-1], color='k', linestyle='--')
        ax[l, m].axvline(x=ths_op[-1], color='k', linestyle='--')
```

```
    if i>0:
        ax[l, m].axvspan(ths_op[0], ths_op[i], facecolor='darkgreen', alpha=0.5)

    #ax[l, m].axvspan(ths_op[i+1], ths_op[j-1], facecolor='lightgreen', alpha=0.5)
    ax[l, m].axvspan(ths_op[j], ths_op[-1], facecolor='lime', alpha=0.5)


if j < i-1:

    ax[l, m].axvline(x=ths_op[0], color='k', linestyle='--')
    ax[l, m].axvline(x=ths_op[i], color='k', linestyle='--')
    ax[l, m].axvline(x=ths_op[i+1], color='k', linestyle='--')
    ax[l, m].axvline(x=ths_op[j+1], color='k', linestyle='--')
    ax[l, m].axvline(x=ths_op[-1], color='k', linestyle='--')

    if i<4:
        ax[l, m].axvspan(ths_op[i+1], ths_op[-1], facecolor='darkgreen', alpha=0.5)

    ax[l, m].axvspan(ths_op[j+1], ths_op[i], facecolor='lightgreen', alpha=0.5)
    ax[l, m].axvspan(ths_op[0], ths_op[j+1], facecolor='lime', alpha=0.5)

if (j < i) & ~(j < i-1):

    ax[l, m].axvline(x=ths_op[0], color='k', linestyle='--')
    ax[l, m].axvline(x=ths_op[i], color='k', linestyle='--')
    ax[l, m].axvline(x=ths_op[i+1], color='k', linestyle='--')
    ax[l, m].axvline(x=ths_op[j+1], color='k', linestyle='--')
    ax[l, m].axvline(x=ths_op[-1], color='k', linestyle='--')

    if i<4:
        ax[l, m].axvspan(ths_op[i+1], ths_op[-1], facecolor='darkgreen', alpha=0.5)

    #ax[l, m].axvspan(ths_op[j+1], ths_op[i], facecolor='lightgreen', alpha=0.5)
    ax[l, m].axvspan(ths_op[0], ths_op[j+1], facecolor='lime', alpha=0.5)

ax[l, m].set_xlim([0.05, 0.95])
ax[l, m].set_ylim([res[i][j].min() - 1 * (res[i][j].max() - res[i][j].min()),
                   res[i][j].max() + 1 * (res[i][j].max() - res[i][j].min())])

ax[l, m].plot(x, res[i][j], color=color_value, marker=marker_value,
              linewidth=3, markersize=12, label=label_value)

ax[l, m].fill_between(x, res_l[i][j], res_r[i][j], color=fill_color_value, alpha=.

#ax[l, m].grid()

ax[l, m].set_xticks(x)
ax[l, m].set_xticklabels(names, fontsize=15)
```

```
def row(i, l, res, res_l, res_r, names):

    title_size_value=25
    label_value = 1

    ##########################################################################################

    for m in range(len(ths_op)-1):

        #m = 0
        j = m
        ax[l, m].set_title(f'{names[i]}->{names[j]}', fontsize=title_size_value)
        color_value='k'
        fill_color_value = 'dimgray'
        marker_value = '*'
        aux_plot(l, m,
                 x, res, res_l, res_r, i, j, color_value, fill_color_value, marker_valu



def aux_plot_rad(x, res, res_l, res_r, i, j,
                 color_value, fill_color_value,
                 marker_value,
                 label_value, subplot):



    subplot.plot(x, res[i][j], color=color_value, marker=marker_value,
                 linewidth=3, markersize=12, label=label_value)

    subplot.fill_between(x, res_l[i][j], res_r[i][j], color=fill_color_value, alpha=.1)




def fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask_):

    res = np.zeros((len(ths_op)-1, len(ths_fr_op)-1))

    for i in range(len(ths_op) - 1):
        for j in range(len(ths_fr_op) - 1):

            mask_slf_pos = (ths_op[i] < data[op1]) & (data[op1] <= ths_op[i+1])
            mask_fr_avg_op = (ths_fr_op[j] < data[fr_op1]) & (data[fr_op1] <= ths_fr_op
            mask = mask_slf_pos & mask_fr_avg_op & mask_

            #print(sum(mask))
```

```
            p = abs(op_change)[mask].mean()

            res[i, j] = p

    return res


def aux_plot_magnitude(title_value, k, subplot, x, ths_op, table1, table2):

    subplot.set_title(title_value, fontsize=30)

    subplot.plot(x, table1[k], color='blue', marker='*', label='neg',
                 linewidth=3, markersize=12)


    subplot.plot(x, table2[k], color='k', marker='o', label='pos',
                 linewidth=3, markersize=12)

    subplot.axvline(x=ths_op[k], color='k', linestyle='--')
    subplot.axvline(x=ths_op[k+1], color='k', linestyle='--')
    subplot.axvspan(ths_op[k], ths_op[k+1], facecolor='lightcoral', alpha=0.5)



def aux_plot_magnitude_control_neg(title_value, k, subplot, x, ths_op, table1, table2)

    subplot.set_title(title_value, fontsize=30)

    subplot.plot(x, table1[k], color='blue', marker='*', #label='$EPOC_{-}$',
                 linewidth=3, markersize=12)


    subplot.plot(x, table2[k], color='purple', marker='o', #label='$EPOC_{+}$',
                 linewidth=3, markersize=12)

    min_value = min(table1[k].min(), table2[k].min()) - 0.05
    max_value = min(table1[k].max(), table2[k].max()) + 0.05

    #subplot.set_ylim([min_value/2, max_value*2])

    subplot.axvline(x=ths_op[k], color='k', linestyle='--')
    subplot.axvline(x=ths_op[k+1], color='k', linestyle='--')
    subplot.axvspan(ths_op[k], ths_op[k+1], facecolor='lightcoral', alpha=0.5)

def aux_plot_EPOC_magnitude_pos(title_value, k, subplot, x, ths_op, table1, table2):
```

```
    subplot.set_title(title_value, fontsize=30)

    subplot.plot(x, table1[k], color='k', marker='*', #label='$EPOC_{-}$',
                 linewidth=3, markersize=12)


    subplot.plot(x, table2[k], color='darkgreen', marker='o', #label='$EPOC_{+}$',
                 linewidth=3, markersize=12)

    min_value = min(table1[k].min(), table2[k].min()) - 0.05
    max_value = min(table1[k].max(), table2[k].max()) + 0.05

    #subplot.set_ylim([min_value/2, max_value*2])

    subplot.axvline(x=ths_op[k], color='k', linestyle='--')
    subplot.axvline(x=ths_op[k+1], color='k', linestyle='--')
    subplot.axvspan(ths_op[k], ths_op[k+1], facecolor='lightcoral', alpha=0.5)



def fill_tables_balance(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask

    res = np.zeros((len(ths_op)-1, len(ths_fr_op)-1))

    for i in range(len(ths_op) - 1):
        for j in range(len(ths_fr_op) - 1):

            mask_slf_pos = (ths_op[i] < data[op1]) & (data[op1] <= ths_op[i+1])
            mask_fr_avg_op = (ths_fr_op[j] < data[fr_op1]) & (data[fr_op1] <= ths_fr_op

            mask_move & mask_pos & control_mask
            mask_move & mask_neg & control_mask

            nom = sum(mask_slf_pos & mask_fr_avg_op & mask_nom)
            denom = sum(mask_slf_pos & mask_fr_avg_op & mask_denom)

            if denom>0:
                p = nom / denom
            else:
                p = None

            res[i, j] = p

    return res


def aux_plot_balance(title_value, k, subplot, x, ths_op, table):
```

```
        subplot.set_title(title_value, fontsize=30)

        subplot.plot(x, table[k], color='k', marker='o',
                     linewidth=3, markersize=12)

        subplot.axvline(x=ths_op[k], color='k', linestyle='--')
        subplot.axvline(x=ths_op[k+1], color='k', linestyle='--')
        subplot.axvspan(ths_op[k], ths_op[k+1], facecolor='lightcoral', alpha=0.5)
```

1 Data preparation

1.1 Download data from the paper of Kozitsin et al. (2019)

```
In [17]: data = pd.read_csv('X_opinions.scv')
         #data.drop([data.columns[0], data.columns[1], data.columns[4], data.columns[6]], axis
         data.drop(['Unnamed: 0'], axis=1, inplace=True)
         data.head()
```

Out[17]:

	$\hat{x}_{i}(t_{1})$	$\hat{x}_{i}(t_{2})$	$\hat{x}_{i}(t_{3})$	\
0	0.514611	0.514624	0.512251	
1	0.186235	0.186235	0.186192	
2	0.604282	0.603247	0.603549	
3	0.608380	0.614673	0.637561	
4	0.729980	0.722895	0.726959	

	$\hat{x}_{-i}(t_{1})$	$\hat{x}_{-i}(t_{2})$	$\hat{x}_{-i}(t_{3})$
0	0.410485	0.467247	0.457429
1	0.544541	0.576840	0.577952
2	0.333697	0.308934	0.286581
3	0.410863	0.413971	0.413348
4	0.605165	0.616444	0.614579

```
In [18]: A = sprs.load_npz('X_friends.npz')

         A
```

```
Out[18]: <1660927x1660927 sparse matrix of type '<class 'numpy.float32'>'
                 with 29036316 stored elements in Compressed Sparse Row format>
```

1.2 Analyze connectivity patterns

```
In [19]: G = nx.from_scipy_sparse_matrix(A)
```

```
In [20]: cc = nx.connected_components(G) #Fing connected components
         list_of_cc = [len(c) for c in sorted(cc, key=len, reverse=True)] #sizes of connected
```

```
In [21]: print('sizes of connected components arranged in the decrease order:')
         print(list_of_cc)
```

```
sizes of connected components arranged in the decrease order:
[1648829, 11, 8, 8, 7, 7, 7, 7, 7, 7, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 5, 5, 5, 5, 5, 5,
```

1.2.1 We observe one giant connected component and many tiny components

```
In [23]: print('Number of connected components:', len(list_of_cc))
         print('Fraction of nodes in the giant connected component:', round(list_of_cc[0] / A.s
```

```
Number of connected components: 5536
Fraction of nodes in the giant connected component: 0.993
```

1.2.2 Hereafter we are focused on the giant connected component

```
In [ ]: largest_cc = max(nx.connected_components(G), key=len)
        largest_cc = np.array(list(largest_cc))
        #np.save('largest_cc.npy', largest_cc)

        data = data[data.index.isin(largest_cc)]
        A = A[largest_cc, :][:, largest_cc]

        #data.to_csv('X_opinions_cc.scv', index=False)
        #sprs.save_npz('X_friends_cc.npz', A)
```

1.3 Add information on friends' opinions diversity

```
In [26]: data = pd.read_csv('X_opinions_cc.scv')
         print(data.shape[0])
         data.head()
```

```
1648829
```

```
Out[26]:    $\hat{x}_{i}(t_{1})$  $\hat{x}_{i}(t_{2})$  $\hat{x}_{i}(t_{3})$  \
         0              0.514611              0.514624              0.512251
         1              0.186235              0.186235              0.186192
         2              0.604282              0.603247              0.603549
         3              0.608380              0.614673              0.637561
         4              0.729980              0.722895              0.726959

            $\hat{x}_{-i}(t_{1})$  $\hat{x}_{-i}(t_{2})$  $\hat{x}_{-i}(t_{3})$
         0               0.410485               0.467247               0.457429
         1               0.544541               0.576840               0.577952
         2               0.333697               0.308934               0.286581
         3               0.410863               0.413971               0.413348
         4               0.605165               0.616444               0.614579
```

```
In [25]: A = sprs.load_npz('X_friends_cc.npz')
         A

Out[25]: <1648829x1648829 sparse matrix of type '<class 'numpy.float32'>'
                 with 29023034 stored elements in Compressed Sparse Row format>

In [29]: n = A.shape[0]

         ops = np.array(data[data.columns[0]])
         diversity1 = []
         for i in tqdm(range(n)):
             fr = get_fr(A, i)
             fr_ops = ops[fr]
             diversity1.append(fr_ops.std())

         ops = np.array(data[data.columns[1]])
         diversity2 = []
         for i in tqdm(range(n)):
             fr = get_fr(A, i)
             fr_ops = ops[fr]
             diversity2.append(fr_ops.std())

         ops = np.array(data[data.columns[2]])
         diversity3 = []
         for i in tqdm(range(n)):
             fr = get_fr(A, i)
             fr_ops = ops[fr]
             diversity3.append(fr_ops.std())
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=1648829.0), HTML(value='')))
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=1648829.0), HTML(value='')))
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=1648829.0), HTML(value='')))
```

1.4 Add information on ideological composition (at time t_2)

```
In [32]: ops = np.array(data[data.columns[1]])
         ths_op = np.arange(0, 1.2, 0.2)

         new_data = []

         for i in tqdm(range(n)):

             new_data_temp = []
             fr = get_fr(A, i)
             op_frs = ops[fr]

             masks = []
             masks.append(op_frs < 0.2)  #Friends Strong Liberals
             masks.append((0.2 <= op_frs) & (op_frs < 0.4))  #Friends Liberals
             masks.append((0.4 <= op_frs) & (op_frs < 0.6))  #Friends Liberals
             masks.append((0.6 <= op_frs) & (op_frs < 0.8))  #Friends Conservatives
             masks.append(0.8 <= op_frs)  #Friends Strong Conservatives

             for mask in masks:
                 new_data_temp.append(len(op_frs[mask]))

             new_data_temp.append(len(fr))

             new_data.append(new_data_temp)

         new_data = np.array(new_data)

         fr_total = np.array(A.sum(axis=1)).ravel()
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=1648829.0), HTML(value='')))
```

1.5 Create pandas table that is used in the current research and save it

```
In [33]: data = pd.DataFrame({'ops1' : data[data.columns[0]],
                               'ops2' : data[data.columns[1]],
                               'ops3' : data[data.columns[2]],
                               'fr_ops1_avg' : data[data.columns[3]],
                               'fr_ops1_std' : diversity1,
                               'fr_ops2_avg' : data[data.columns[4]],
                               'fr_ops2_std' : diversity2,
                               'fr_ops3_avg' : data[data.columns[5]],
                               'fr_ops3_std' : diversity3,
                               'fr_number' : fr_total,
```

```
                                    'SL-Friends' : new_data[:, 0],
                                    'L-Friends' : new_data[:, 1],
                                    'M-Friends' : new_data[:, 2],
                                    'C-Friends' : new_data[:, 3],
                                    'SC-Friends' : new_data[:, 4]})

        #data.to_csv('main_data.csv', index=False)

        data.head()
```

Out[33]:

	ops1	ops2	ops3	fr_ops1_avg	fr_ops1_std	fr_ops2_avg	fr_ops2_std	fr_ops3_avg	fr_ops3_std	fr_number	SL-Friends	L-Friends	M-Friends	C-Friends	SC-Friends
0	0.514611	0.514624	0.512251	0.410485	0.261048	0.467247	0.251042	0.457429	0.252795	7.0	2	0	3	1	1
1	0.186235	0.186235	0.186192	0.544541	0.227728	0.576840	0.238795	0.577952	0.239873	7.0	1	0	1	5	0
2	0.604282	0.603247	0.603549	0.333697	0.167857	0.308934	0.144153	0.286581	0.132653	6.0	2	2	2	0	0
3	0.608380	0.614673	0.637561	0.410863	0.269629	0.413971	0.264415	0.413348	0.264016	3.0	1	0	1	1	0
4	0.729980	0.722895	0.726959	0.605165	0.108791	0.616444	0.104925	0.614579	0.106953	3.0	0	0	2	1	0

2 Download data

```
In [2]: data = pd.read_csv('main_data.csv')

        n = data.shape[0]

        print('number of users:', n)

        data.head()
```

```
number of users: 1648829
```

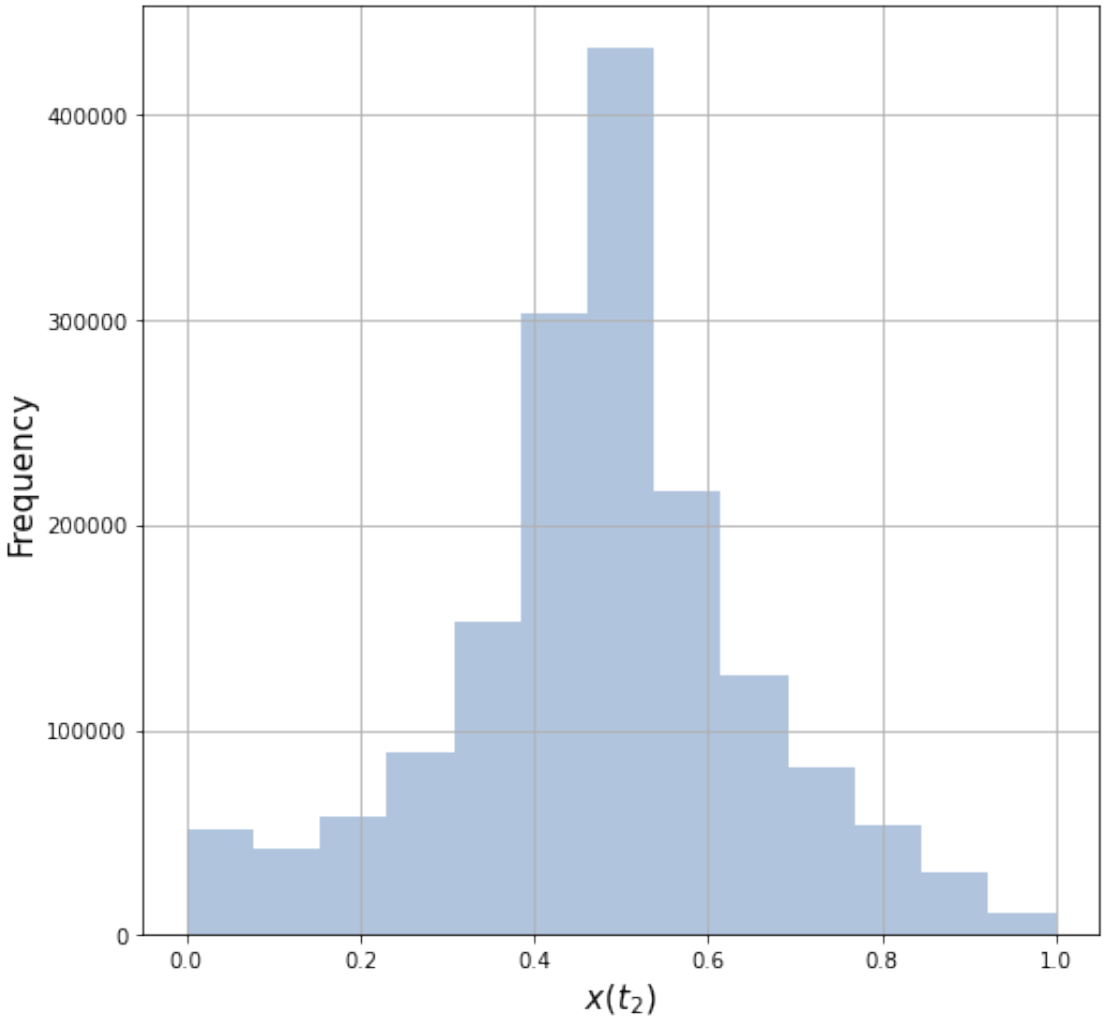
Out[2]:

	ops1	ops2	ops3	fr_ops1_avg	fr_ops1_std	fr_ops2_avg
0	0.514611	0.514624	0.512251	0.410485	0.261048	0.467247
1	0.186235	0.186235	0.186192	0.544541	0.227728	0.576840
2	0.604282	0.603247	0.603549	0.333697	0.167857	0.308934

```
3  0.608380  0.614673  0.637561     0.410863     0.269629     0.413971
4  0.729980  0.722895  0.726959     0.605165     0.108791     0.616444

   fr_ops2_std  fr_ops3_avg  fr_ops3_std  fr_number  SL-Friends  L-Friends  \
0     0.251042     0.457429     0.252795        7.0           2          0
1     0.238795     0.577952     0.239873        7.0           1          0
2     0.144153     0.286581     0.132653        6.0           2          2
3     0.264415     0.413348     0.264016        3.0           1          0
4     0.104925     0.614579     0.106953        3.0           0          0

   M-Friends  C-Friends  SC-Friends
0          3          1           1
1          1          5           0
2          2          0           0
3          1          1           0
4          2          1           0
```

3 Information on ideological groups

In [3]:
```
t = 'ops2'
ths_op = np.arange(0, 1.2, 0.2) #[0, 0.35, 0.45, 0.55, 0.65, 1]

plt.figure(figsize=(8, 8))
data[t].hist(bins=13, color='lightsteelblue')
plt.ylabel('Frequency', fontsize=15)
plt.xlabel('$x(t_2)$', fontsize=15)
plt.show()
```



3.1 Table 2

```
In [10]: mask_SL = data[t] < ths_op[1]
         mask_L = (data[t] >= ths_op[1]) & (data[t] < ths_op[2])
         mask_M = (data[t] >= ths_op[2]) & (data[t] < ths_op[3])
         mask_C = (data[t] >= ths_op[3]) & (data[t] < ths_op[4])
         mask_SC = (data[t] >= ths_op[4])

         print('Number of SL:', sum(mask_SL), ', avg number of friends:', data[mask_SL]['fr_nu
         print('Number of L:', sum(mask_L), ', avg number of friends:', data[mask_L]['fr_numbe
         print('Number of M:', sum(mask_M), ', avg number of friends:', data[mask_M]['fr_numbe
         print('Number of C:', sum(mask_C), ', avg number of friends:', data[mask_C]['fr_numbe
         print('Number of SC:', sum(mask_SC), ', avg number of friends:', data[mask_SC]['fr_nu

         print('Correlation between opinion and number of friends:', data[[t, 'fr_number']].co
```

```
Number of SL: 125714 , avg number of friends: 25.681435639626454
Number of L: 312063 , avg number of friends: 21.201289483213326
Number of M: 874327 , avg number of friends: 17.12569896617627
Number of C: 266509 , avg number of friends: 13.028700719300286
Number of SC: 70216 , avg number of friends: 10.434274239489575
Correlation between opinion and number of friends: -0.16296246502055012
```

4 Homophily patterns

4.1 Table 5

```
In [14]: data['SL-Friends rel'] = data['SL-Friends'] / data['fr_number']
         data['L-Friends rel'] = data['L-Friends'] / data['fr_number']
         data['M-Friends rel'] = data['M-Friends'] / data['fr_number']
         data['C-Friends rel'] = data['C-Friends'] / data['fr_number']
         data['SC-Friends rel'] = data['SC-Friends'] / data['fr_number']

         control_mask = 1 == 1
         i = -5
         res = np.zeros((5, 5))
         res[0] = np.round(np.array(data[mask_SL & control_mask][data.columns[i:]].mean(axis=0)
         res[1] = np.round(np.array(data[mask_L & control_mask][data.columns[i:]].mean(axis=0))
         res[2] = np.round(np.array(data[mask_M & control_mask][data.columns[i:]].mean(axis=0))
         res[3] = np.round(np.array(data[mask_C & control_mask][data.columns[i:]].mean(axis=0))
         res[4] = np.round(np.array(data[mask_SC & control_mask][data.columns[i:]].mean(axis=0)
         pd.DataFrame(res, index=['SLs', 'Ls', 'Ms', 'Cs', 'SCs'], columns=['SL-friends',
                                                                            'L-friends',
                                                                            'M-friends',
                                                                            'C-friends',
                                                                            'SC-friends'])
```

Out[14]:

	SL-friends	L-friends	M-friends	C-friends	SC-friends
SL	0.15	0.24	0.46	0.11	0.03
L	0.11	0.23	0.51	0.12	0.03
M	0.08	0.20	0.54	0.15	0.03
C	0.07	0.17	0.49	0.21	0.06
SC	0.07	0.15	0.45	0.24	0.10

4.2 Table C1

```
In [15]: control_mask = data['fr_number'] < 5
         i = -5
         res = np.zeros((5, 5))
         res[0] = np.round(np.array(data[mask_SL & control_mask][data.columns[i:]].mean(axis=0)
         res[1] = np.round(np.array(data[mask_L & control_mask][data.columns[i:]].mean(axis=0))
         res[2] = np.round(np.array(data[mask_M & control_mask][data.columns[i:]].mean(axis=0))
         res[3] = np.round(np.array(data[mask_C & control_mask][data.columns[i:]].mean(axis=0))
```

```
res[4] = np.round(np.array(data[mask_SC & control_mask][data.columns[i:]].mean(axis=0)
pd.DataFrame(res, index=['SLs', 'Ls', 'Ms', 'Cs', 'SCs'], columns=['SL-friends',
                                                                   'L-friends',
                                                                   'M-friends',
                                                                   'C-friends',
                                                                   'SC-friends'])
```

Out[15]:

	SL-friends	L-friends	M-friends	C-friends	SC-friends
SLs	0.14	0.22	0.44	0.15	0.05
Ls	0.09	0.21	0.51	0.15	0.04
Ms	0.07	0.18	0.53	0.18	0.05
Cs	0.06	0.15	0.46	0.24	0.09
SCs	0.06	0.13	0.40	0.27	0.14

4.3 Table C2

```
In [16]: control_mask = (data['fr_number'] >= 5) & (data['fr_number'] <= 25)
         i = -5
         res = np.zeros((5, 5))
         res[0] = np.round(np.array(data[mask_SL & control_mask][data.columns[i:]].mean(axis=0)
         res[1] = np.round(np.array(data[mask_L & control_mask][data.columns[i:]].mean(axis=0)
         res[2] = np.round(np.array(data[mask_M & control_mask][data.columns[i:]].mean(axis=0)
         res[3] = np.round(np.array(data[mask_C & control_mask][data.columns[i:]].mean(axis=0)
         res[4] = np.round(np.array(data[mask_SC & control_mask][data.columns[i:]].mean(axis=0)
         pd.DataFrame(res, index=['SLs', 'Ls', 'Ms', 'Cs', 'SCs'], columns=['SL-friends',
                                                                            'L-friends',
                                                                            'M-friends',
                                                                            'C-friends',
                                                                            'SC-friends'])
```

Out[16]:

	SL-friends	L-friends	M-friends	C-friends	SC-friends
SLs	0.15	0.24	0.47	0.12	0.03
Ls	0.10	0.22	0.52	0.13	0.03
Ms	0.08	0.20	0.54	0.15	0.03
Cs	0.07	0.17	0.50	0.20	0.05
SCs	0.07	0.16	0.47	0.23	0.08

4.4 Table C3

```
In [18]: control_mask = data['fr_number'] > 25
         i = -5
         res = np.zeros((5, 5))
         res[0] = np.round(np.array(data[mask_SL & control_mask][data.columns[i:]].mean(axis=0)
         res[1] = np.round(np.array(data[mask_L & control_mask][data.columns[i:]].mean(axis=0)
         res[2] = np.round(np.array(data[mask_M & control_mask][data.columns[i:]].mean(axis=0)
         res[3] = np.round(np.array(data[mask_C & control_mask][data.columns[i:]].mean(axis=0)
         res[4] = np.round(np.array(data[mask_SC & control_mask][data.columns[i:]].mean(axis=0)
         pd.DataFrame(res, index=['SLs', 'Ls', 'Ms', 'Cs', 'SCs'], columns=['SL-friends',
```

```
                                                                'L-friends',
                                                                'M-friends',
                                                                'C-friends',
                                                                'SC-friends'])
```

Out[18]:

	SL-friends	L-friends	M-friends	C-friends	SC-friends
SLs	0.17	0.27	0.47	0.08	0.02
Ls	0.13	0.26	0.50	0.09	0.02
Ms	0.11	0.23	0.54	0.11	0.02
Cs	0.09	0.20	0.53	0.15	0.03
SCs	0.08	0.20	0.51	0.17	0.04

5 Opinion dynamics analysis

In [43]:
```
print('Number of individuals with opinion x=0.5')

print('Time moment t1:', sum(data['ops1'] == 0.5))
print('Time moment t2:', sum(data['ops2'] == 0.5))
print('Time moment t3:', sum(data['ops3'] == 0.5))

print('\n')

print('Time step t1->t2')

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m


print('Number of remarkable movements:', sum(mask_move))

print('Proportion of positive movements among remarkable ones:',
      round(sum(mask_move & mask_pos) / sum(mask_move), 3))

print('Proportion of skipping remarkable movements among positive remarkable:',
      round(sum(mask_skip) / sum(mask_pos), 3))

print("How diversity of friends' opinions varies between skipping and non-skipping mov
      "Mean diversity of skipping movements:", round(data[mask_skip][fr_op1_divergence
      "Mean diversity of non-skipping movements:", round(data[mask_no_skip][fr_op1_div

print("Number of users whose friends' average opinion changed for no more than 0.05:"
      round(sum(mask_control_fr) / n, 3))

print('Number of remarkable movements exclued since they do not meet the previous filt
      round(sum(mask_move & ~mask_control_fr) / sum(mask_move), 3))
```

```
print('Map of movements (Table 5 part 1):')
movs = np.zeros((len(masks_now), len(masks_future)))
for i in range(len(masks_now)):
    for j in range(len(masks_future)):
        movs[i, j] = sum(masks_now[i] & masks_future[j])

movs = movs.astype(int)
map_of_moves = pd.DataFrame(movs,
             columns=['->SL', '->L', '->M', '->C', '->SC'],
             index=['SL', 'L', 'M', 'C', 'SC'])

n_classes = 5
incomes = np.zeros(n_classes)
outcomes = np.zeros(n_classes)
for i in range(n_classes):
    outcomes[i] = movs[i, np.arange(n_classes)[np.arange(n_classes) != i]].sum()
    incomes[i] = movs[np.arange(n_classes)[np.arange(n_classes) != i], i].sum()
map_of_moves['Population growth'] = incomes
map_of_moves['Population decrease'] = outcomes
map_of_moves['Relative growth'] = round((map_of_moves['Population growth'] - map_of_mo
print(tabulate(map_of_moves, headers='keys', tablefmt='psql'))

print('\n')

print('Time step t2->t3')

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m


print('Number of remarkable movements:', sum(mask_move))

print('Proportion of positive movements among remarkable ones:',
      round(sum(mask_move & mask_pos) / sum(mask_move), 3))

print('Proportion of skipping remarkable movements among positive remarkable:',
      round(sum(mask_skip) / sum(mask_pos), 3))

print("How diversity of friends' opinions varies between skipping and non-skipping mov
      "Mean diversity of skipping movements:", round(data[mask_skip][fr_op1_divergence
      "Mean diversity of non-skipping movements:", round(data[mask_no_skip][fr_op1_div

print("Number of users whose friends' average opinion changed for no more than 0.05:"
      round(sum(mask_control_fr) / n, 3))

print('Number of remarkable movements exclued since they do not meet the previous filt
      round(sum(mask_move & ~mask_control_fr) / sum(mask_move), 3))
```

```
        print('Map of movements (Table 5 part 2):')
        movs = np.zeros((len(masks_now), len(masks_future)))
        for i in range(len(masks_now)):
            for j in range(len(masks_future)):
                movs[i, j] = sum(masks_now[i] & masks_future[j])

        movs = movs.astype(int)
        map_of_moves = pd.DataFrame(movs,
                     columns=['->SL', '->L', '->M', '->C', '->SC'],
                     index=['SL', 'L', 'M', 'C', 'SC'])

        n_classes = 5
        incomes = np.zeros(n_classes)
        outcomes = np.zeros(n_classes)
        for i in range(n_classes):
            outcomes[i] = movs[i, np.arange(n_classes)[np.arange(n_classes) != i]].sum()
            incomes[i] = movs[np.arange(n_classes)[np.arange(n_classes) != i], i].sum()
        map_of_moves['Population growth'] = incomes
        map_of_moves['Population decrease'] = outcomes
        map_of_moves['Relative growth'] = round((map_of_moves['Population growth'] - map_of_mo
        print(tabulate(map_of_moves, headers='keys', tablefmt='psql'))
```

```
Number of individuals with opinion x=0.5
Time moment t1: 0
Time moment t2: 0
Time moment t3: 0


Time step t1->t2
Number of remarkable movements: 147344
Proportion of positive movements among remarkable ones: 0.602
Proportion of skipping remarkable movements among positive remarkable: 0.099
How diversity of friends' opinions varies between skipping and non-skipping movements:  Mean di
Number of users whose friends' average opinion changed for no more than 0.05: 0.977
Number of remarkable movements exclued since they do not meet the previous filter: 0.032
Map of movements (Table 5 part 1):
+----+--------+--------+--------+--------+--------+---------------------+---------------------
|    |   ->SL |    ->L |    ->M |    ->C |   ->SC |   Population growth |   Population decreas
|----+--------+--------+--------+--------+--------+---------------------+---------------------
| SL | 113081 |   5662 |    991 |    117 |     25 |               12633 |                 679
| L  |  11141 | 269961 |  21032 |    448 |     49 |               42102 |                3267
| M  |   1307 |  35724 | 830633 |  16706 |    233 |               43694 |                5397
| C  |    147 |    636 |  21245 | 244116 |   4863 |               22393 |                2689
| SC |     38 |     80 |    426 |   5122 |  65046 |                5170 |                 566
+----+--------+--------+--------+--------+--------+---------------------+---------------------
```

```
Time step t2->t3
Number of remarkable movements: 84210
Proportion of positive movements among remarkable ones: 0.582
Proportion of skipping remarkable movements among positive remarkable: 0.07
How diversity of friends' opinions varies between skipping and non-skipping movements:  Mean di
Number of users whose friends' average opinion changed for no more than 0.05: 0.988
Number of remarkable movements exclued since they do not meet the previous filter: 0.019
Map of movements (Table 5 part 2):
+----+--------+--------+--------+--------+--------+---------------------+---------------------
|    |   ->SL |    ->L |    ->M |    ->C |   ->SC |   Population growth |   Population decrease
|----+--------+--------+--------+--------+--------+---------------------+---------------------
| SL | 121036 |   3915 |    645 |     98 |     20 |                8865 |                 4678
| L  |   8071 | 291304 |  12483 |    186 |     19 |               30433 |                20759
| M  |    690 |  26107 | 835230 |  12194 |    106 |               27909 |                39097
| C  |     83 |    351 |  14454 | 247695 |   3926 |               15983 |                18814
| SC |     21 |     60 |    327 |   3505 |  66303 |                4071 |                 3913
+----+--------+--------+--------+--------+--------+---------------------+---------------------
```

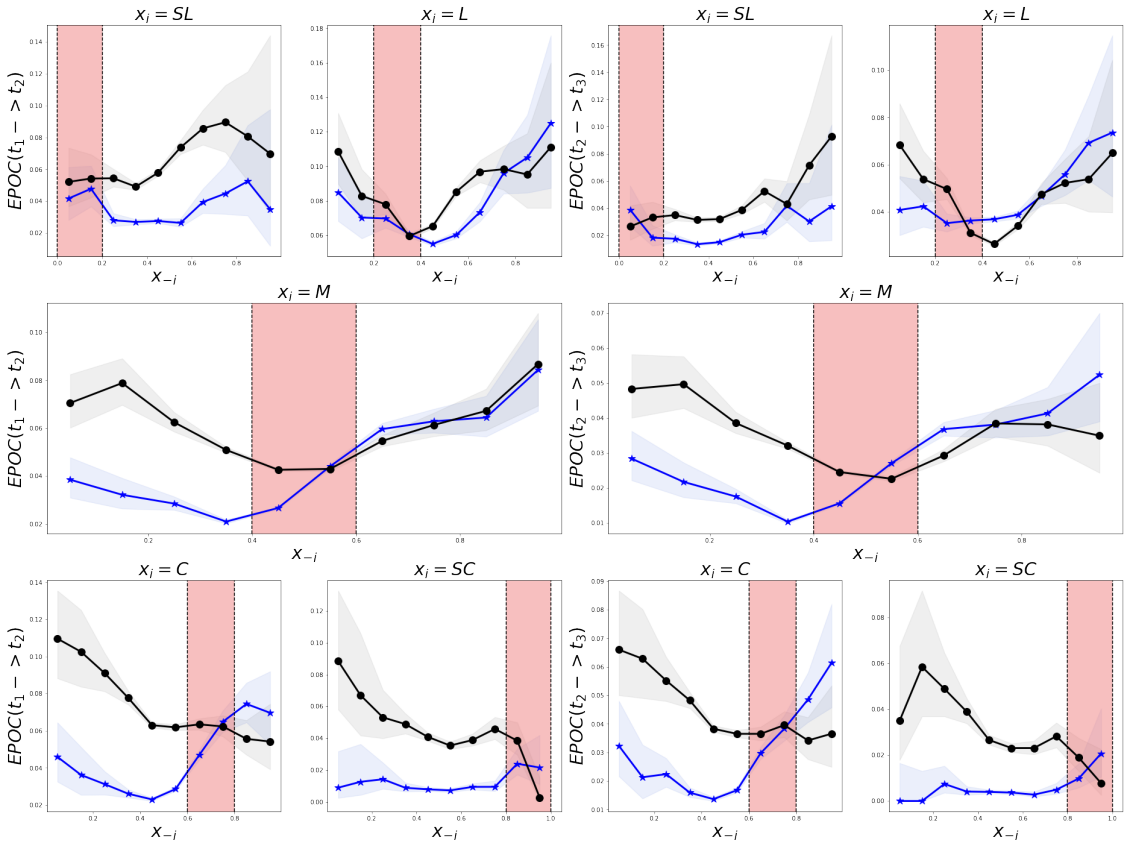
6 $EPOC_+, EPOC_-$ as a function of x_{-i}, control for x_i (Figure 4)

```
In [4]: ths_op = np.arange(0, 1.2, 0.2)
        ths_fr_op = np.arange(0, 1.1, 0.1)

        ##############################################################################################

        t1 = 1
        t2 = 2
        op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

        #EPOC neg

        mask_nom = mask_move & mask_neg & mask_control_fr
        mask_denom = mask_control_fr

        tables1 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

        #EPOC pos

        mask_nom = mask_move & mask_pos & mask_control_fr
        mask_denom = mask_control_fr

        tables2 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

        ##############################################################################################

        t1 = 2
```

```
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#EPOC neg

mask_nom = mask_move & mask_neg & mask_control_fr
mask_denom = mask_control_fr

tables3 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_fr
mask_denom = mask_control_fr

tables4 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)


fig = plt.figure(figsize=(32, 24))

#plt.title('A', fontsize=50)
#plt.text(0, 0, 'A', fontsize=50)

xlabel = "$x_{-i}$"
ylabel1 = '$EPOC(t_{1} -> t_{2})$'
ylabel2 = '$EPOC(t_{2} -> t_{3})$'
x = ths_fr_op[1:] - 0.05

#########################

sub1 = fig.add_subplot(3, 4, 1) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC(title_value, k, sub1, x, ths_op, tables1, tables2)

#########################

sub2 = fig.add_subplot(3, 4, 2) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC(title_value, k, sub2, x, ths_op, tables1, tables2)

#########################

sub3 = fig.add_subplot(3, 4, 3) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC(title_value, k, sub3, x, ths_op, tables3, tables4)
```

```
########################

sub4 = fig.add_subplot(3, 4, 4) # 3 rows, 4 columns, 4 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC(title_value, k, sub4, x, ths_op, tables3, tables4)

########################

sub5 = fig.add_subplot(3, 4, (5,6))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC(title_value, k, sub5, x, ths_op, tables1, tables2)

########################

sub6 = fig.add_subplot(3, 4, (7,8))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC(title_value, k, sub6, x, ths_op, tables3, tables4)

########################

sub7 = fig.add_subplot(3, 4, 9)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC(title_value, k, sub7, x, ths_op, tables1, tables2)

########################

sub8 = fig.add_subplot(3, 4, 10)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC(title_value, k, sub8, x, ths_op, tables1, tables2)

########################

sub9 = fig.add_subplot(3, 4, 11)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC(title_value, k, sub9, x, ths_op, tables3, tables4)

########################

sub10 = fig.add_subplot(3, 4, 12)
k = 4
title_value = '$x_{i}=SC$'
```

```
aux_plot_EPOC(title_value, k, sub10, x, ths_op, tables3, tables4)

########################

#sub1.legend(fontsize=15, loc='upper left')
#sub2.legend(fontsize=15, loc='upper left')
#sub3.legend(fontsize=15, loc='upper center')
#sub4.legend(fontsize=15, loc='upper left')
#sub5.legend(fontsize=15, loc='upper right')

size = 30

#for sub in [sub1, sub2, sub3, sub4, sub5, sub6, sub7, sub8, sub9, sub10]:

#    sub.set_ylim([0, 0.13])

sub1.set_xlabel(xlabel, fontsize=size)
sub2.set_xlabel(xlabel, fontsize=size)
sub3.set_xlabel(xlabel, fontsize=size)
sub4.set_xlabel(xlabel, fontsize=size)
sub5.set_xlabel(xlabel, fontsize=size)
sub6.set_xlabel(xlabel, fontsize=size)
sub7.set_xlabel(xlabel, fontsize=size)
sub8.set_xlabel(xlabel, fontsize=size)
sub9.set_xlabel(xlabel, fontsize=size)
sub10.set_xlabel(xlabel, fontsize=size)

sub1.set_ylabel(ylabel1, fontsize=size)
sub3.set_ylabel(ylabel2, fontsize=size)
sub5.set_ylabel(ylabel1, fontsize=size)
sub6.set_ylabel(ylabel2, fontsize=size)
sub7.set_ylabel(ylabel1, fontsize=size)
sub9.set_ylabel(ylabel2, fontsize=size)

plt.show()
```



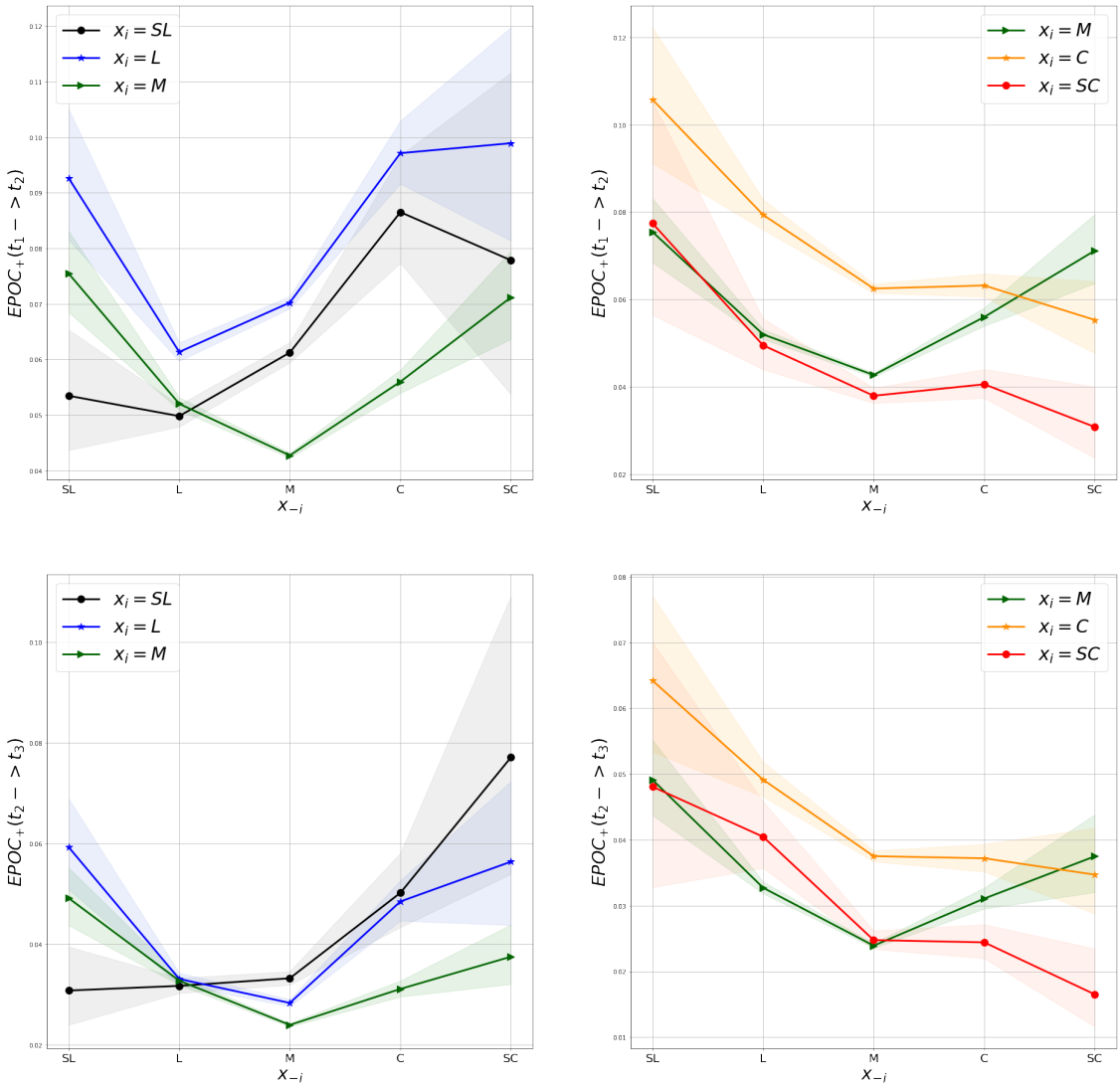
6.0.1 EPOC as a function of friends' average opinion x_i across ideological groups, separated by movement type. Colored columns represent the value of x_i. Black dots plot $EPOC_+$; blue stars plot $EPOC_-$.

7 Compare $EPOC_+$ across ideological groups (Figure C2)

```
In [6]: ths_op = np.arange(0, 1.2, 0.2)
        ths_fr_op = np.arange(0, 1.2, 0.2)

        ##########################################################################################

        t1 = 1
        t2 = 2
        op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

        #EPOC neg

        mask_nom = mask_move & mask_neg & mask_control_fr
        mask_denom = mask_control_fr
```

```
tables1 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_fr
mask_denom = mask_control_fr

tables2 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

##############################################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#EPOC neg

mask_nom = mask_move & mask_neg & mask_control_fr
mask_denom = mask_control_fr

tables3 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_fr
mask_denom = mask_control_fr

tables4 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)


fig = plt.figure(figsize=(32, 32))

#plt.title('A', fontsize=50)
#plt.text(0, 0, 'A', fontsize=50)

xlabel = "$x_{-i}$"
ylabel1 = '$EPOC_{+}(t_{1} -> t_{2})$'
ylabel2 = '$EPOC_{+}(t_{2} -> t_{3})$'
x = ths_fr_op[1:] - 0.1

########################

sub1 = fig.add_subplot(2, 2, 1) # 2 rows, 2 columns, 1 cell
aux_plot_EPOC_plus_compare_SL_M(sub1, x, tables2)
########################

sub2 = fig.add_subplot(2, 2, 2) # 2 rows, 2 columns, 2 cell
aux_plot_EPOC_plus_compare_M_SC(sub2, x, tables2)
```

```
########################

sub3 = fig.add_subplot(2, 2, 3) # 2 rows, 2 columns, 3 cell
aux_plot_EPOC_plus_compare_SL_M(sub3, x, tables4)
########################

sub4 = fig.add_subplot(2, 2, 4) # 2 rows, 2 columns, 4 cell
aux_plot_EPOC_plus_compare_M_SC(sub4, x, tables4)

########################

sub1.legend(fontsize=30, loc='upper left')
sub2.legend(fontsize=30)
sub3.legend(fontsize=30, loc='upper left')
sub4.legend(fontsize=30)

label_size = 30
tick_size = 20

sub1.set_xlabel(xlabel, fontsize=label_size)
sub1.set_xticks(x)
sub1.set_xticklabels(['SL', 'L', 'M', 'C', 'SC'], fontsize=tick_size)
sub2.set_xlabel(xlabel, fontsize=label_size)
sub2.set_xticks(x)
sub2.set_xticklabels(['SL', 'L', 'M', 'C', 'SC'], fontsize=tick_size)
sub3.set_xlabel(xlabel, fontsize=label_size)
sub3.set_xticks(x)
sub3.set_xticklabels(['SL', 'L', 'M', 'C', 'SC'], fontsize=tick_size)
sub4.set_xlabel(xlabel, fontsize=label_size)
sub4.set_xticks(x)
sub4.set_xticklabels(['SL', 'L', 'M', 'C', 'SC'], fontsize=tick_size)

sub1.set_ylabel(ylabel1, fontsize=label_size)
sub2.set_ylabel(ylabel1, fontsize=label_size)
sub3.set_ylabel(ylabel2, fontsize=label_size)
sub4.set_ylabel(ylabel2, fontsize=label_size)



plt.show()
```



7.0.1 $EPOC_+$ **as a function of** x_{-i}**, separated by** x_i**. We emphasize the following regularity. For two groups** G_1 **and** G_2**, where without loss of generality** $G_1 < G_2$**, it tends to hold that $ EPOC_{+} (x_{i}=G_{1},x_{-i}=G_{2})<EPOC_{+} (x_{i}=G_{2},x_{-i}=G_{1}) $ if** G_2M **and** $EPOC_+(x_i = G_1, x_{-i} = G_2) > EPOC_+(x_i = G_2, x_{-i} = G_1)$ **if** G_1M**. Exceptions here are: (1) Ls and Ms for time step $ t_{1}t_{2};$ Ms and Cs for both time steps.**

8 How EPOC measures are affected by the value of $|x_{-i}(t_{k+1}) - x_{-i}(t_k)|$ (Figure C1)

```
In [7]: ths_op = np.arange(0, 1.2, 0.2)
        ths_fr_op = np.arange(0, 1.1, 0.1)

        # fr' average opinion shift is less than 0.05
```

```
##############################################################################
##############################################################################

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#EPOC neg

mask_nom = mask_move & mask_neg & mask_control_fr
mask_denom = mask_control_fr

tables1 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_fr
mask_denom = mask_control_fr

tables2 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

##############################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#EPOC neg

mask_nom = mask_move & mask_neg & mask_control_fr
mask_denom = mask_control_fr

tables3 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_fr
mask_denom = mask_control_fr

tables4 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

# fr' average opinion shift is greator than or equal to 0.05
##############################################################################
##############################################################################

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma
```

```
#EPOC neg

mask_nom = mask_move & mask_neg & ~mask_control_fr
mask_denom = ~mask_control_fr

tables5 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & ~mask_control_fr
mask_denom = ~mask_control_fr

tables6 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

##############################################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#EPOC neg

mask_nom = mask_move & mask_neg & ~mask_control_fr
mask_denom = ~mask_control_fr

tables7 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & ~mask_control_fr
mask_denom = ~mask_control_fr

tables8 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)


fig = plt.figure(figsize=(32, 48))

#plt.title('A', fontsize=50)
#plt.text(0, 0, 'A', fontsize=50)

xlabel = "$x_{-i}$"
ylabel11 = '$EPOC_{-}(t_{1} -> t_{2})$'
ylabel12 = '$EPOC_{-}(t_{2} -> t_{3})$'
ylabel21 = '$EPOC_{+}(t_{1} -> t_{2})$'
ylabel22 = '$EPOC_{+}(t_{2} -> t_{3})$'
x = ths_fr_op[1:] - 0.05
```

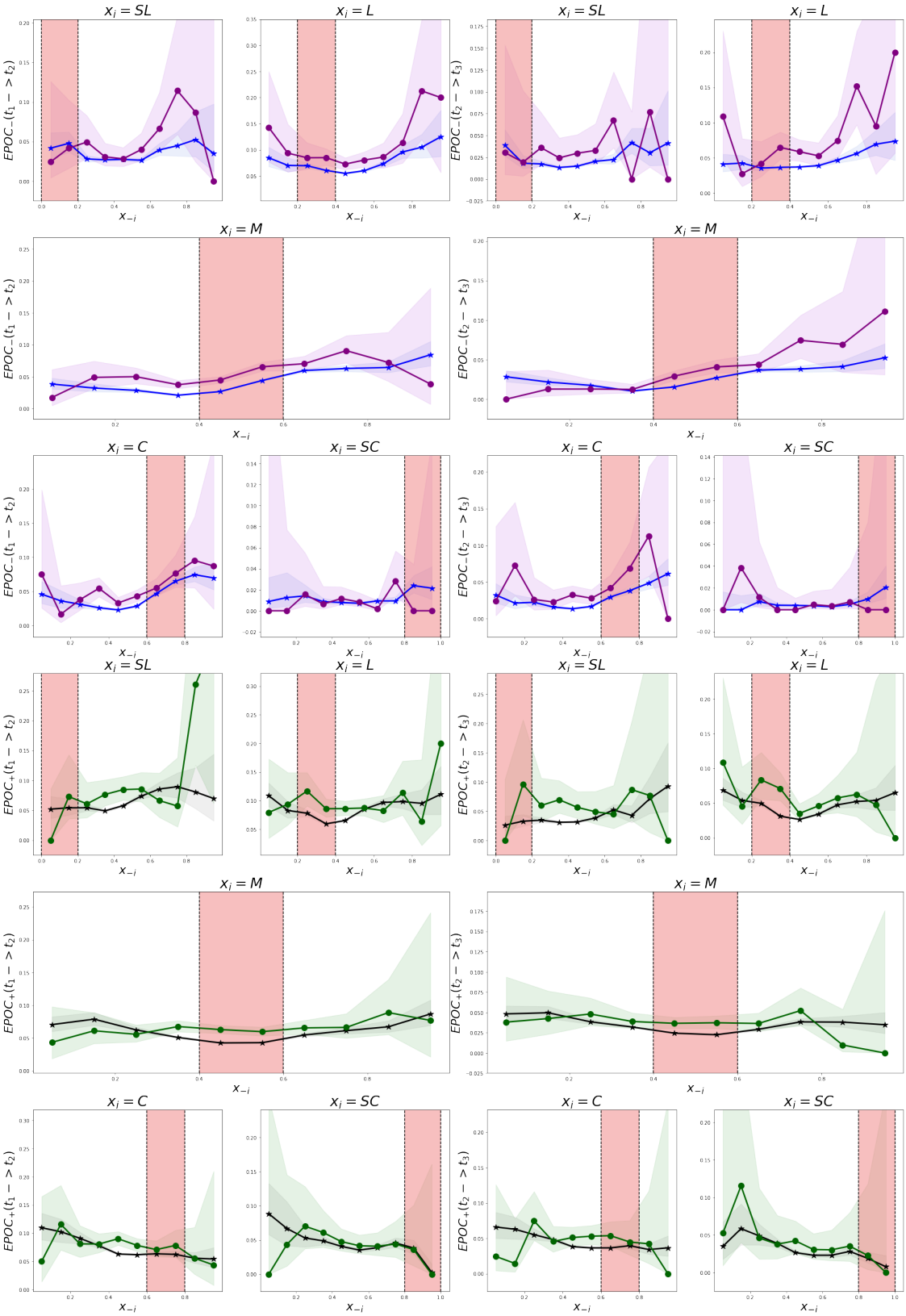
```
#########################

sub1 = fig.add_subplot(6, 4, 1) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_control_neg(title_value, k, sub1, x, ths_op, tables1, tables5)

#########################

sub2 = fig.add_subplot(6, 4, 2) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_control_neg(title_value, k, sub2, x, ths_op, tables1, tables5)

#########################

sub3 = fig.add_subplot(6, 4, 3) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_control_neg(title_value, k, sub3, x, ths_op, tables3, tables7)

#########################

sub4 = fig.add_subplot(6, 4, 4) # 3 rows, 4 columns, 4 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_control_neg(title_value, k, sub4, x, ths_op, tables3, tables7)

#########################

sub5 = fig.add_subplot(6, 4, (5,6))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_control_neg(title_value, k, sub5, x, ths_op, tables1, tables5)

#########################

sub6 = fig.add_subplot(6, 4, (7,8))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_control_neg(title_value, k, sub6, x, ths_op, tables3, tables7)

#########################

sub7 = fig.add_subplot(6, 4, 9)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_control_neg(title_value, k, sub7, x, ths_op, tables1, tables5)
```

```
########################

sub8 = fig.add_subplot(6, 4, 10)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_control_neg(title_value, k, sub8, x, ths_op, tables1, tables5)

########################

sub9 = fig.add_subplot(6, 4, 11)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_control_neg(title_value, k, sub9, x, ths_op, tables3, tables7)

########################

sub10 = fig.add_subplot(6, 4, 12)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_control_neg(title_value, k, sub10, x, ths_op, tables3, tables7)

#########################################################################
#########################################################################

sub11 = fig.add_subplot(6, 4, 13) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_control_pos(title_value, k, sub11, x, ths_op, tables2, tables6)

########################

sub12 = fig.add_subplot(6, 4, 14) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_control_pos(title_value, k, sub12, x, ths_op, tables2, tables6)

########################

sub13 = fig.add_subplot(6, 4, 15) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_control_pos(title_value, k, sub13, x, ths_op, tables4, tables8)

########################

sub14 = fig.add_subplot(6, 4, 16) # 3 rows, 4 columns, 4 cell
k = 1
```

```
title_value = '$x_{i}=L$'
aux_plot_EPOC_control_pos(title_value, k, sub14, x, ths_op, tables4, tables8)

########################

sub15 = fig.add_subplot(6, 4, (17,18))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_control_pos(title_value, k, sub15, x, ths_op, tables2, tables6)

########################

sub16 = fig.add_subplot(6, 4, (19,20))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_control_pos(title_value, k, sub16, x, ths_op, tables4, tables8)

########################

sub17 = fig.add_subplot(6, 4, 21)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_control_pos(title_value, k, sub17, x, ths_op, tables2, tables6)

########################

sub18 = fig.add_subplot(6, 4, 22)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_control_pos(title_value, k, sub18, x, ths_op, tables2, tables6)

########################

sub19 = fig.add_subplot(6, 4, 23)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_control_pos(title_value, k, sub19, x, ths_op, tables4, tables8)

########################

sub20 = fig.add_subplot(6, 4, 24)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_control_pos(title_value, k, sub20, x, ths_op, tables4, tables8)

#############################################################################
#############################################################################
```

```
#sub1.legend(fontsize=15, loc='upper left')
#sub2.legend(fontsize=15, loc='upper left')
#sub3.legend(fontsize=15, loc='upper center')
#sub4.legend(fontsize=15, loc='upper left')
#sub5.legend(fontsize=15, loc='upper right')

size = 25

#for sub in [sub1, sub2, sub3, sub4, sub5, sub6, sub7, sub8, sub9, sub10]:

#    sub.set_ylim([0, 0.13])

sub1.set_xlabel(xlabel, fontsize=size)
sub2.set_xlabel(xlabel, fontsize=size)
sub3.set_xlabel(xlabel, fontsize=size)
sub4.set_xlabel(xlabel, fontsize=size)
sub5.set_xlabel(xlabel, fontsize=size)
sub6.set_xlabel(xlabel, fontsize=size)
sub7.set_xlabel(xlabel, fontsize=size)
sub8.set_xlabel(xlabel, fontsize=size)
sub9.set_xlabel(xlabel, fontsize=size)
sub10.set_xlabel(xlabel, fontsize=size)

sub1.set_ylabel(ylabel11, fontsize=size)
sub3.set_ylabel(ylabel12, fontsize=size)
sub5.set_ylabel(ylabel11, fontsize=size)
sub6.set_ylabel(ylabel12, fontsize=size)
sub7.set_ylabel(ylabel11, fontsize=size)
sub9.set_ylabel(ylabel12, fontsize=size)

sub11.set_xlabel(xlabel, fontsize=size)
sub12.set_xlabel(xlabel, fontsize=size)
sub13.set_xlabel(xlabel, fontsize=size)
sub14.set_xlabel(xlabel, fontsize=size)
sub15.set_xlabel(xlabel, fontsize=size)
sub16.set_xlabel(xlabel, fontsize=size)
sub17.set_xlabel(xlabel, fontsize=size)
sub18.set_xlabel(xlabel, fontsize=size)
sub19.set_xlabel(xlabel, fontsize=size)
sub20.set_xlabel(xlabel, fontsize=size)

sub11.set_ylabel(ylabel21, fontsize=size)
sub13.set_ylabel(ylabel22, fontsize=size)
sub15.set_ylabel(ylabel21, fontsize=size)
sub16.set_ylabel(ylabel22, fontsize=size)
sub17.set_ylabel(ylabel21, fontsize=size)
sub19.set_ylabel(ylabel22, fontsize=size)
```

```
plt.show()
```


$x_i = SL$
$x_i = L$
$x_i = SL$
$x_i = L$
$EPOC_-(t_1 -> t_2)$
$EPOC_-(t_2 -> t_3)$
x_{-i}
$x_i = M$
$x_i = M$
$EPOC_-(t_1 -> t_2)$
$EPOC_-(t_2 -> t_3)$
x_{-i}
$x_i = C$
$x_i = SC$
$x_i = C$
$x_i = SC$
$EPOC_-(t_1 -> t_2)$
$EPOC_-(t_2 -> t_3)$
x_{-i}
$x_i = SL$
$x_i = L$
$x_i = SL$
$x_i = L$
$EPOC_+(t_1 -> t_2)$
$EPOC_+(t_2 -> t_3)$
x_{-i}
$x_i = M$
$x_i = M$
$EPOC_+(t_1 -> t_2)$
$EPOC_+(t_2 -> t_3)$
x_{-i}
$x_i = C$
$x_i = SC$
$x_i = C$
$x_i = SC$
$EPOC_+(t_1 -> t_2)$
$EPOC_+(t_2 -> t_3)$
x_{-i}

8.0.1 In this plot, we demonstrate how $EPOC(t_k t_{k+1})$ is moderated by the value of $|x_{-i}(t_{k+1}) - x_{-i}(t_k)|$. Blue stars plot $EPOC_{-}(t_k t_{k+1})$ if $|x_{-i}(t_{k+1}) - x_{-i}(t_k)|$ is under 0.05; purple dots plot $EPOC_{-}(t_k t_{k+1})$ if $|x_{-i}(t_{k+1}) - x_{-i}(t_k)|0.05$. Black stars plot $EPOC_{+}(t_k t_{k+1})$ if $|x_{-i}(t_{k+1}) - x_{-i}(t_k)|$ is under 0.05; green dots plot $EPOC_{+}(t_k t_{k+1})$ if $|x_{-i}(t_{k+1}) - x_{-i}(t_k)|0.05$. Higher values of $|x_{-i}(t_{k+1}) - x_{-i}(t_k)|$ are correlated with a higher chance of an opinion shift.

9 How EPOC measures are affected by the values of diversity of friends' opinions σ_{-i} (Figure is not included in the main manuscript)

In [8]:
```
ths_op = np.arange(0, 1.2, 0.2)
ths_fr_op = np.arange(0, 1.1, 0.1)

# fr' average opinion shift is less than 0.05
##############################################################################################
##############################################################################################

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

mask_control = (data[fr_op1_divergence] < 0.15)  & mask_control_fr
mask_control_ = (data[fr_op1_divergence] >= 0.15)  & mask_control_fr

#EPOC neg

mask_nom = mask_move & mask_neg & mask_control
mask_denom = mask_control

tables1 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control
mask_denom = mask_control

tables2 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

##############################################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma
```

```
#EPOC neg

mask_nom = mask_move & mask_neg & mask_control
mask_denom = mask_control

tables3 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control
mask_denom = mask_control

tables4 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

# fr' average opinion shift is greator than or equal to 0.05
##############################################################################
##############################################################################

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#EPOC neg

mask_nom = mask_move & mask_neg & mask_control_
mask_denom = mask_control_

tables5 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_
mask_denom = mask_control_

tables6 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)

##############################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#EPOC neg

mask_nom = mask_move & mask_neg & mask_control_
mask_denom = mask_control_

tables7 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)
```

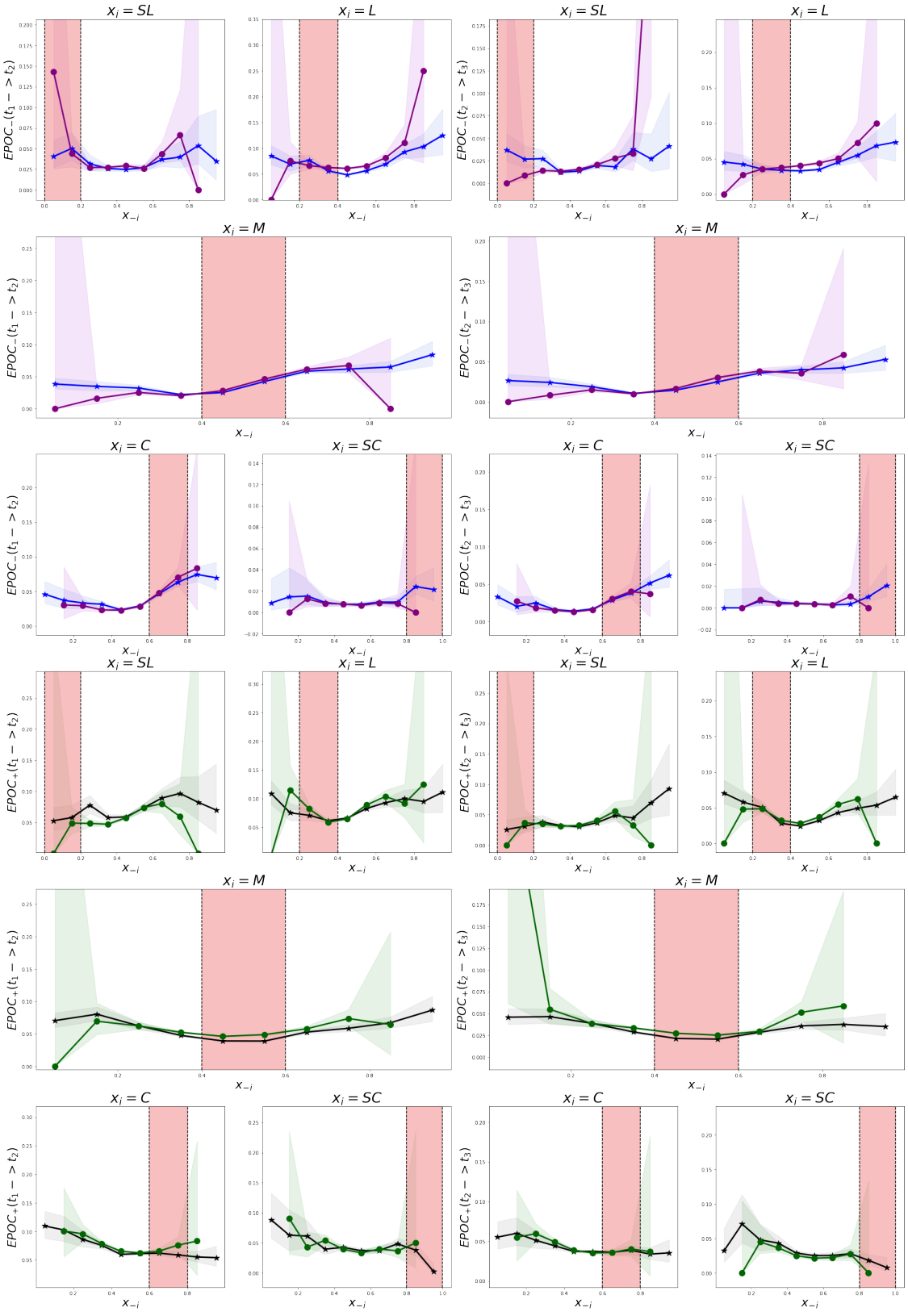
```
#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_
mask_denom = mask_control_

tables8 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom)


fig = plt.figure(figsize=(32, 48))

#plt.title('A', fontsize=50)
#plt.text(0, 0, 'A', fontsize=50)

xlabel = "$x_{-i}$"
ylabel11 = '$EPOC_{-}(t_{1} -> t_{2})$'
ylabel12 = '$EPOC_{-}(t_{2} -> t_{3})$'
ylabel21 = '$EPOC_{+}(t_{1} -> t_{2})$'
ylabel22 = '$EPOC_{+}(t_{2} -> t_{3})$'
x = ths_fr_op[1:] - 0.05

#########################

sub1 = fig.add_subplot(6, 4, 1) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_control_neg(title_value, k, sub1, x, ths_op, tables1, tables5)

#########################

sub2 = fig.add_subplot(6, 4, 2) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_control_neg(title_value, k, sub2, x, ths_op, tables1, tables5)

#########################

sub3 = fig.add_subplot(6, 4, 3) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_control_neg(title_value, k, sub3, x, ths_op, tables3, tables7)

#########################

sub4 = fig.add_subplot(6, 4, 4) # 3 rows, 4 columns, 4 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_control_neg(title_value, k, sub4, x, ths_op, tables3, tables7)
```

```
########################

sub5 = fig.add_subplot(6, 4, (5,6))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_control_neg(title_value, k, sub5, x, ths_op, tables1, tables5)

########################

sub6 = fig.add_subplot(6, 4, (7,8))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_control_neg(title_value, k, sub6, x, ths_op, tables3, tables7)

########################

sub7 = fig.add_subplot(6, 4, 9)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_control_neg(title_value, k, sub7, x, ths_op, tables1, tables5)

########################

sub8 = fig.add_subplot(6, 4, 10)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_control_neg(title_value, k, sub8, x, ths_op, tables1, tables5)

########################

sub9 = fig.add_subplot(6, 4, 11)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_control_neg(title_value, k, sub9, x, ths_op, tables3, tables7)

########################

sub10 = fig.add_subplot(6, 4, 12)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_control_neg(title_value, k, sub10, x, ths_op, tables3, tables7)

#########################################################################
#########################################################################

sub11 = fig.add_subplot(6, 4, 13) # 3 rows, 4 columns, 1 cell
k = 0
```

```
title_value = '$x_{i}=SL$'
aux_plot_EPOC_control_pos(title_value, k, sub11, x, ths_op, tables2, tables6)

########################

sub12 = fig.add_subplot(6, 4, 14) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_control_pos(title_value, k, sub12, x, ths_op, tables2, tables6)

########################

sub13 = fig.add_subplot(6, 4, 15) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_control_pos(title_value, k, sub13, x, ths_op, tables4, tables8)

########################

sub14 = fig.add_subplot(6, 4, 16) # 3 rows, 4 columns, 4 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_control_pos(title_value, k, sub14, x, ths_op, tables4, tables8)

########################

sub15 = fig.add_subplot(6, 4, (17,18))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_control_pos(title_value, k, sub15, x, ths_op, tables2, tables6)

########################

sub16 = fig.add_subplot(6, 4, (19,20))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_control_pos(title_value, k, sub16, x, ths_op, tables4, tables8)

########################

sub17 = fig.add_subplot(6, 4, 21)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_control_pos(title_value, k, sub17, x, ths_op, tables2, tables6)

########################

sub18 = fig.add_subplot(6, 4, 22)
```

```
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_control_pos(title_value, k, sub18, x, ths_op, tables2, tables6)

########################

sub19 = fig.add_subplot(6, 4, 23)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_control_pos(title_value, k, sub19, x, ths_op, tables4, tables8)

########################

sub20 = fig.add_subplot(6, 4, 24)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_control_pos(title_value, k, sub20, x, ths_op, tables4, tables8)

#########################################################################
#########################################################################

#sub1.legend(fontsize=15, loc='upper left')
#sub2.legend(fontsize=15, loc='upper left')
#sub3.legend(fontsize=15, loc='upper center')
#sub4.legend(fontsize=15, loc='upper left')
#sub5.legend(fontsize=15, loc='upper right')

size = 25

#for sub in [sub1, sub2, sub3, sub4, sub5, sub6, sub7, sub8, sub9, sub10]:

#    sub.set_ylim([0, 0.13])

sub1.set_xlabel(xlabel, fontsize=size)
sub2.set_xlabel(xlabel, fontsize=size)
sub3.set_xlabel(xlabel, fontsize=size)
sub4.set_xlabel(xlabel, fontsize=size)
sub5.set_xlabel(xlabel, fontsize=size)
sub6.set_xlabel(xlabel, fontsize=size)
sub7.set_xlabel(xlabel, fontsize=size)
sub8.set_xlabel(xlabel, fontsize=size)
sub9.set_xlabel(xlabel, fontsize=size)
sub10.set_xlabel(xlabel, fontsize=size)

sub1.set_ylabel(ylabel11, fontsize=size)
sub3.set_ylabel(ylabel12, fontsize=size)
sub5.set_ylabel(ylabel11, fontsize=size)
sub6.set_ylabel(ylabel12, fontsize=size)
```

```
sub7.set_ylabel(ylabel11, fontsize=size)
sub9.set_ylabel(ylabel12, fontsize=size)

sub11.set_xlabel(xlabel, fontsize=size)
sub12.set_xlabel(xlabel, fontsize=size)
sub13.set_xlabel(xlabel, fontsize=size)
sub14.set_xlabel(xlabel, fontsize=size)
sub15.set_xlabel(xlabel, fontsize=size)
sub16.set_xlabel(xlabel, fontsize=size)
sub17.set_xlabel(xlabel, fontsize=size)
sub18.set_xlabel(xlabel, fontsize=size)
sub19.set_xlabel(xlabel, fontsize=size)
sub20.set_xlabel(xlabel, fontsize=size)

sub11.set_ylabel(ylabel21, fontsize=size)
sub13.set_ylabel(ylabel22, fontsize=size)
sub15.set_ylabel(ylabel21, fontsize=size)
sub16.set_ylabel(ylabel22, fontsize=size)
sub17.set_ylabel(ylabel21, fontsize=size)
sub19.set_ylabel(ylabel22, fontsize=size)

plt.show()
```


$x_i = SL$
$x_i = L$
$x_i = SL$
$x_i = L$
$EPOC_-(t_1 -> t_2)$
$EPOC_-(t_2 -> t_3)$
x_{-i}
$x_i = M$
$x_i = M$
$x_i = C$
$x_i = SC$
$x_i = C$
$x_i = SC$
$x_i = SL$
$x_i = L$
$x_i = SL$
$x_i = L$
$EPOC_+(t_1 -> t_2)$
$EPOC_+(t_2 -> t_3)$
$x_i = M$
$x_i = M$
$x_i = C$
$x_i = SC$
$x_i = C$
$x_i = SC$

9.0.1 In this plot, we demonstrate how $EPOC(t_k t_{k+1})$ is moderated by the value of σ_{-i}. Blue stars plot $EPOC_{-}(t_k t_{k+1})$ if σ_{-i} is under 0.15; purple dots plot $EPOC_{-}(t_k t_{k+1})$ if σ_{-i}0.15. Black stars plot $EPOC_{+}(t_k t_{k+1})$ if σ_{-i} is under 0.15; green dots plot $EPOC_{+}(t_k t_{k+1})$ if σ_{-i}0.05. We observe no clear trends. Perhaps, higher values of σ_{-i} are associated with a higher chance of opinion shift, but this tendency is quite weak.

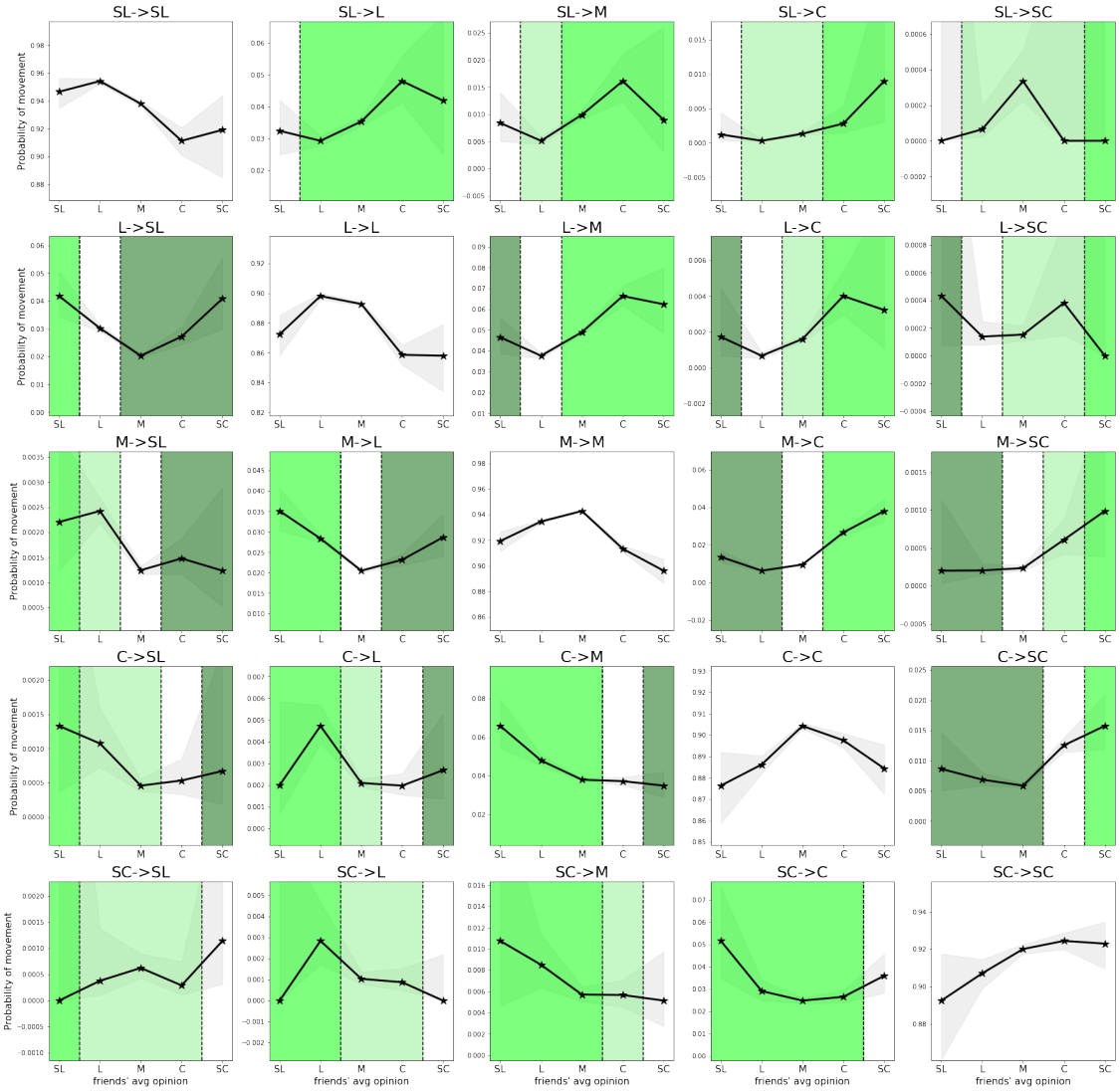
10 All cross-group movements as functions of x_{-i} (Figure B1 and Figure that is not included in the main manuscript)

10.1 Time step t1 -> t2 (Figure B1)

In [9]:
```
ths_op = np.arange(0, 1.2, 0.2)
ths_fr_op = np.arange(0, 1.2, 0.2)

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

control_mask = mask_control_fr
res1, res1_l, res1_r = cross_group_movs_probs(masks_now, masks_future, masks_fr, contro

fig, ax = plt.subplots(5, 5, figsize=(30, 30))

names = ['SL', 'L', 'M', 'C', 'SC']
x = np.arange(0, 1.2, 0.2)[1:] - 0.1 #[0.17, 0.4, 0.5, 0.6, 0.83]
#x = ['SL', 'L', 'C', 'SC']

xlabel = "friends' avg opinion"
ylabel = 'Probability of movement'

##############################################################################
##############################################################################

i = 0
row(i, i, res1, res1_l, res1_r, names)

i = 1
row(i, i, res1, res1_l, res1_r, names)

i = 2
row(i, i, res1, res1_l, res1_r, names)

i = 3
row(i, i, res1, res1_l, res1_r, names)

i = 4
row(i, i, res1, res1_l, res1_r, names)
```

```
        ax[0, 0].set_ylabel(ylabel, fontsize=15)
        ax[1, 0].set_ylabel(ylabel, fontsize=15)
        ax[2, 0].set_ylabel(ylabel, fontsize=15)
        ax[3, 0].set_ylabel(ylabel, fontsize=15)
        ax[4, 0].set_ylabel(ylabel, fontsize=15)

        ax[4, 0].set_xlabel(xlabel, fontsize=15)
        ax[4, 1].set_xlabel(xlabel, fontsize=15)
        ax[4, 2].set_xlabel(xlabel, fontsize=15)
        ax[4, 3].set_xlabel(xlabel, fontsize=15)
        ax[4, 4].set_xlabel(xlabel, fontsize=15)
        ###########################################################################################
        ###########################################################################################


        plt.show()
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=5.0), HTML(value='')))
```


SL->SL
SL->L
SL->M
SL->C
SL->SC
L->SL
L->L
L->M
L->C
L->SC
M->SL
M->L
M->M
M->C
M->SC
C->SL
C->L
C->M
C->C
C->SC
SC->SL
SC->L
SC->M
SC->C
SC->SC
Probability of movement
friends' avg opinion
SL
L
M
C
SC

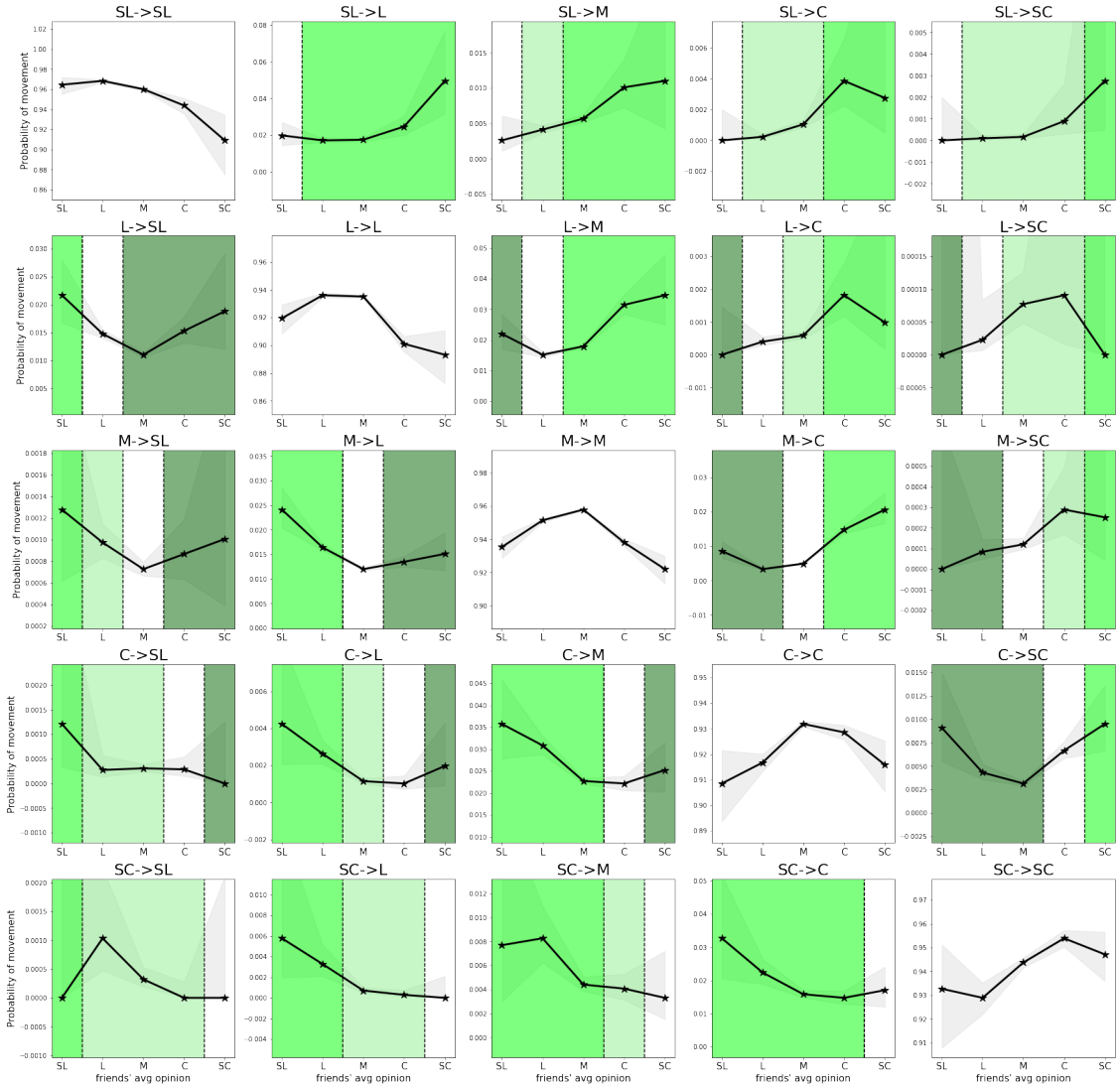
10.1.1 These subplots represent how the probability of a particular opinion shift depends on the average opinion of friends (here, we consider time step t_1t_2; the same subplots for time step t_2t_3 are included in the online supplementary materials). The dashed lines illustrate the boundaries between areas of negative (dark green), positive skipping (light green) and positive nonskipping (lime green) movements. Some subplots demonstrate that the curves do change their behavior after transitioning from the area of skipping movements to the area of nonskipping movements (see, for example, SLC or SCM whereby the curves start to grow with a higher rate after entering the nonskipping zone). However, the similar behavior (growth with an increasing rate) may be observed for movements that imply no skips (CM, SCC).

10.2 Time step t2 -> t3

```
In [10]: ths_op = np.arange(0, 1.2, 0.2)
         ths_fr_op = np.arange(0, 1.2, 0.2)

         t1 = 2
         t2 = 3
         op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

         control_mask = mask_control_fr
         res2, res2_l, res2_r = cross_group_movs_probs(masks_now, masks_future, masks_fr, contr

         fig, ax = plt.subplots(5, 5, figsize=(30, 30))

         names = ['SL', 'L', 'M', 'C', 'SC']
         x = np.arange(0, 1.2, 0.2)[1:] - 0.1
         #x = ['SL', 'L', 'C', 'SC']

         xlabel = "friends' avg opinion"
         ylabel = 'Probability of movement'

         ##########################################################################################
         ##########################################################################################

         i = 0
         row(i, i, res2, res2_l, res2_r, names)

         i = 1
         row(i, i, res2, res2_l, res2_r, names)

         i = 2
         row(i, i, res2, res2_l, res2_r, names)

         i = 3
         row(i, i, res2, res2_l, res2_r, names)

         i = 4
```

```
        row(i, i, res2, res2_l, res2_r, names)

        ax[0, 0].set_ylabel(ylabel, fontsize=15)
        ax[1, 0].set_ylabel(ylabel, fontsize=15)
        ax[2, 0].set_ylabel(ylabel, fontsize=15)
        ax[3, 0].set_ylabel(ylabel, fontsize=15)
        ax[4, 0].set_ylabel(ylabel, fontsize=15)

        ax[4, 0].set_xlabel(xlabel, fontsize=15)
        ax[4, 1].set_xlabel(xlabel, fontsize=15)
        ax[4, 2].set_xlabel(xlabel, fontsize=15)
        ax[4, 3].set_xlabel(xlabel, fontsize=15)
        ax[4, 4].set_xlabel(xlabel, fontsize=15)
        ###############################################################################
        ###############################################################################


        plt.show()
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=5.0), HTML(value='')))
```

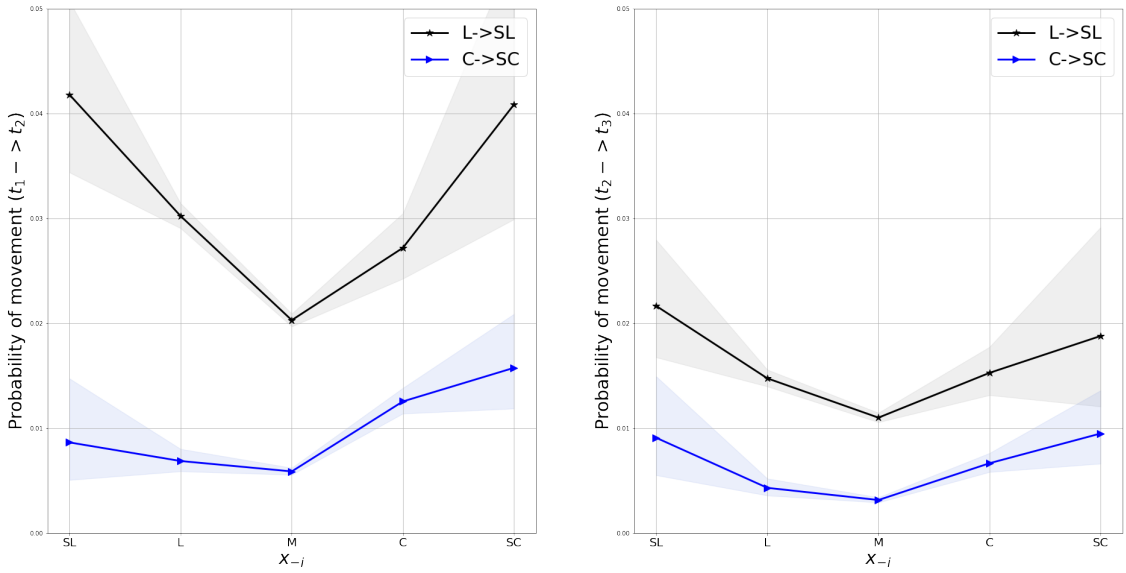


11 Radicalization (Figure 5)

```
In [11]: fig = plt.figure(figsize=(32, 16))

         names = ['SL', 'L', 'M', 'C', 'SC']
         xlabel = "$x_{-i}$"
         ylabel1 = '$EPOC_{+}(t_{1} -> t_{2})$'
         ylabel2 = '$EPOC_{+}(t_{2} -> t_{3})$'
         x = np.arange(0, 1.2, 0.2)[1:] - 0.1

         ########################

         sub1 = fig.add_subplot(1, 2, 1) # 2 rows, 2 columns, 1 cell
```

```
sub1.grid()
sub1.set_ylim([0, 0.05])

i = 1
j = 0
color_value='k'
fill_color_value = 'dimgray'
marker_value = '*'
label_value = f'{names[i]}->{names[j]}'
aux_plot_rad(x, res1, res1_l, res1_r, i, j, color_value, fill_color_value, marker_valu

##########################################################################################

i = 3
j = 4
color_value='b'
fill_color_value = 'royalblue'
marker_value = '>'
label_value = f'{names[i]}->{names[j]}'
aux_plot_rad(x, res1, res1_l, res1_r, i, j, color_value, fill_color_value, marker_valu

##########################################################################################
##########################################################################################

sub2 = fig.add_subplot(1, 2, 2) # 2 rows, 2 columns, 2 cell
sub2.grid()
sub2.set_ylim([0, 0.05])

i = 1
j = 0
color_value='k'
fill_color_value = 'dimgray'
marker_value = '*'
label_value = f'{names[i]}->{names[j]}'
aux_plot_rad(x, res2, res2_l, res2_r, i, j, color_value, fill_color_value, marker_valu

##########################################################################################

i = 3
j = 4
color_value='b'
fill_color_value = 'royalblue'
marker_value = '>'
label_value = f'{names[i]}->{names[j]}'
aux_plot_rad(x, res2, res2_l, res2_r, i, j, color_value, fill_color_value, marker_valu
```

```
sub1.legend(fontsize=30)
sub1.set_ylabel('Probability of movement $(t_{1}->t_{2})$', fontsize=30)
sub1.set_xlabel("$x_{-i}$", fontsize=30)
sub1.set_xticks(x)
sub1.set_xticklabels(names, fontsize=20)

sub2.legend(fontsize=30)
sub2.set_ylabel('Probability of movement $(t_{2}->t_{3})$', fontsize=30)
sub2.set_xlabel("$x_{-i}$", fontsize=30)
sub2.set_xticks(x)
sub2.set_xticklabels(names, fontsize=20)

plt.show()
```



11.0.1 Probabilities of movements *LSL* and *CSC* as functions of x_{-i}. The left panel represents time step t_1t_2; the right panel represents time step t_2t_3.

12 Control for σ_{-i} (Figure C3)

In [12]:
```
ths_op = np.arange(0, 1.2, 0.2)
ths_fr_op = np.arange(0, 1.2, 0.2)

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m
```

```
control_mask = (data[fr_op1_divergence] <= 0.15) & mask_control_fr
tables1 = cross_group_movs_probs(masks_now, masks_future, masks_fr, control_mask, mask


control_mask = (data[fr_op1_divergence] > 0.15) & mask_control_fr
tables2 = cross_group_movs_probs(masks_now, masks_future, masks_fr, control_mask, mask

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

control_mask = (data[fr_op1_divergence] <= 0.15) & mask_control_fr
tables3 = cross_group_movs_probs(masks_now, masks_future, masks_fr, control_mask, mask


control_mask = (data[fr_op1_divergence] > 0.15) & mask_control_fr
tables4 = cross_group_movs_probs(masks_now, masks_future, masks_fr, control_mask, mask


fig = plt.figure(figsize=(32, 32))

names = ['SL', 'L', 'M', 'C', 'SC']
xlabel = "$x_{-i}$"
ylabel1 = '$EPOC_{+}(t_{1} -> t_{2})$'
ylabel2 = '$EPOC_{+}(t_{2} -> t_{3})$'
x = np.arange(0, 1.2, 0.2)[1:] - 0.1

################################################################################

sub1 = fig.add_subplot(2, 2, 1) # 2 rows, 2 columns, 1 cell
sub1.grid()
sub1.set_ylim([0, 0.05])



i = 1
j = 0
title_value = f'{names[i]}->{names[j]}'
sub1.set_title(title_value, fontsize=30)

color_value='k'
fill_color_value = 'dimgray'
marker_value = '*'
label_value = '$\sigma_{-i} < 0.15$'
sub1.set_title(title_value, fontsize=30)
aux_plot_rad(x, tables1[0], tables1[1], tables1[2],
             i, j, color_value, fill_color_value, marker_value, label_value, sub1)
```

```
i = 1
j = 0
color_value='darkgreen'
fill_color_value = 'green'
marker_value = 'o'
label_value = '$\sigma_{-i} \geq 0.15$'
aux_plot_rad(x, tables2[0], tables2[1], tables2[2],
             i, j, color_value, fill_color_value, marker_value, label_value, sub1)

##############################################################################

sub2 = fig.add_subplot(2, 2, 2) # 2 rows, 2 columns, 1 cell
sub2.grid()
sub2.set_ylim([0, 0.02])

i = 3
j = 4
title_value = f'{names[i]}->{names[j]}'
sub2.set_title(title_value, fontsize=30)

color_value='blue'
fill_color_value = 'royalblue'
marker_value = '*'
label_value = '$\sigma_{-i} < 0.15$'
aux_plot_rad(x, tables1[0], tables1[1], tables1[2],
             i, j, color_value, fill_color_value, marker_value, label_value, sub2)


i = 3
j = 4
color_value='purple'
fill_color_value = 'darkviolet'
marker_value = 'o'
label_value = '$\sigma_{-i} \geq 0.15$'
aux_plot_rad(x, tables2[0], tables2[1], tables2[2],
             i, j, color_value, fill_color_value, marker_value, label_value, sub2)

##############################################################################

sub3 = fig.add_subplot(2, 2, 3) # 2 rows, 2 columns, 1 cell
sub3.grid()
sub3.set_ylim([0, 0.03])

i = 1
j = 0
title_value = f'{names[i]}->{names[j]}'
sub3.set_title(title_value, fontsize=30)
```

```
color_value='k'
fill_color_value = 'dimgray'
marker_value = '*'
label_value = '$\sigma_{-i} < 0.15$'
aux_plot_rad(x, tables3[0], tables3[1], tables3[2],
             i, j, color_value, fill_color_value, marker_value, label_value, sub3)


i = 1
j = 0
color_value='darkgreen'
fill_color_value = 'green'
marker_value = 'o'
label_value = '$\sigma_{-i} \geq 0.15$'
aux_plot_rad(x, tables4[0], tables4[1], tables4[2],
             i, j, color_value, fill_color_value, marker_value, label_value, sub3)

##############################################################################

sub4 = fig.add_subplot(2, 2, 4) # 2 rows, 2 columns, 1 cell
sub4.grid()
sub4.set_ylim([0, 0.01])

i = 3
j = 4
title_value = f'{names[i]}->{names[j]}'
sub4.set_title(title_value, fontsize=30)

color_value='blue'
fill_color_value = 'royalblue'
marker_value = '*'
label_value = '$\sigma_{-i} < 0.15$'
aux_plot_rad(x, tables3[0], tables3[1], tables3[2],
             i, j, color_value, fill_color_value, marker_value, label_value, sub4)


i = 3
j = 4
color_value='purple'
fill_color_value = 'darkviolet'
marker_value = 'o'
label_value = '$\sigma_{-i} \geq 0.15$'
aux_plot_rad(x, tables4[0], tables4[1], tables4[2],
             i, j, color_value, fill_color_value, marker_value, label_value, sub4)

##############################################################################
```

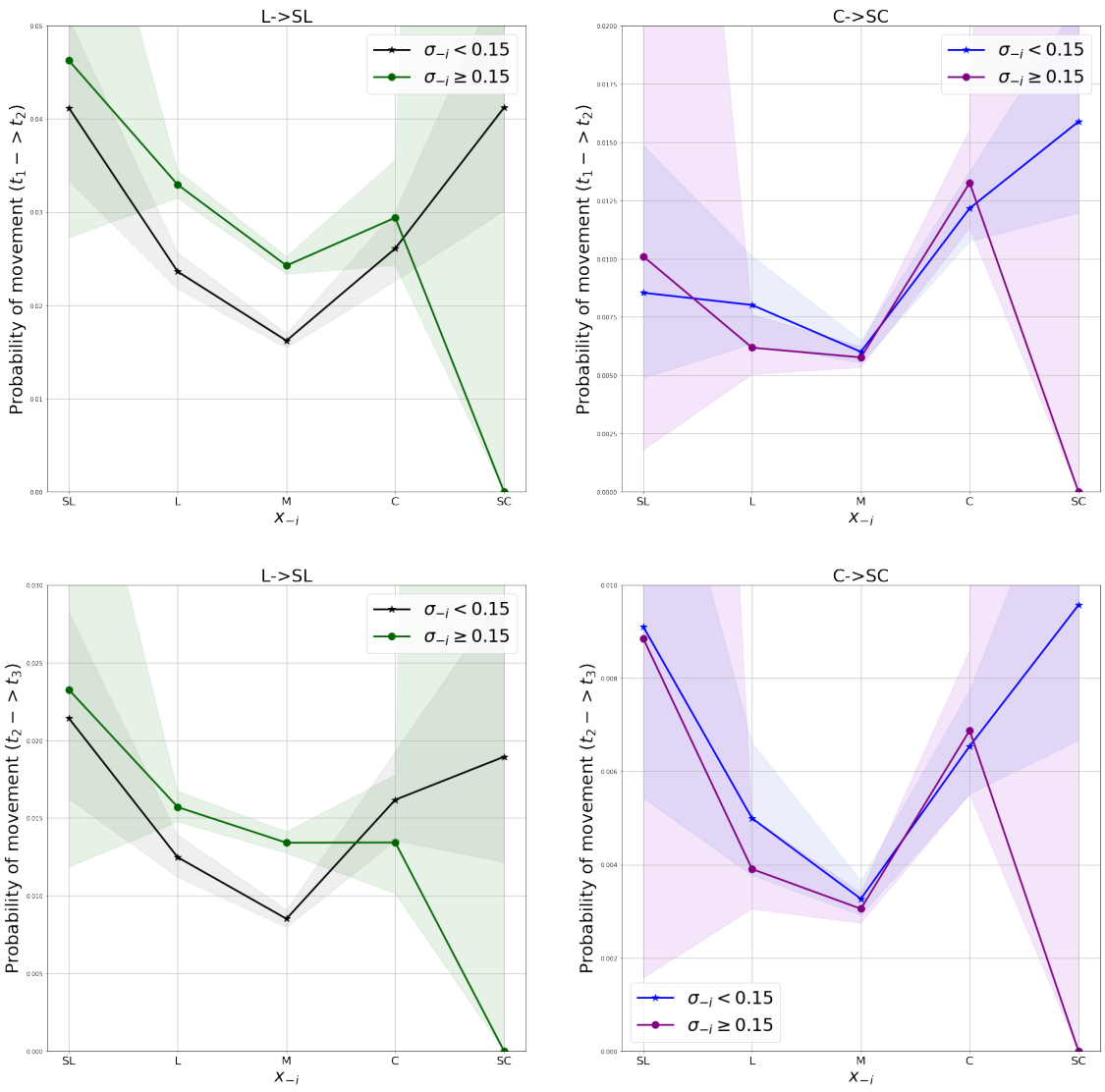
```
        sub1.legend(fontsize=30)
        sub1.set_ylabel('Probability of movement $(t_{1}->t_{2})$', fontsize=30)
        sub1.set_xlabel("$x_{-i}$", fontsize=30)
        sub1.set_xticks(x)
        sub1.set_xticklabels(names, fontsize=20)

        sub2.legend(fontsize=30)
        sub2.set_ylabel('Probability of movement $(t_{1}->t_{2})$', fontsize=30)
        sub2.set_xlabel("$x_{-i}$", fontsize=30)
        sub2.set_xticks(x)
        sub2.set_xticklabels(names, fontsize=20)

        sub3.legend(fontsize=30)
        sub3.set_ylabel('Probability of movement $(t_{2}->t_{3})$', fontsize=30)
        sub3.set_xlabel("$x_{-i}$", fontsize=30)
        sub3.set_xticks(x)
        sub3.set_xticklabels(names, fontsize=20)

        sub4.legend(fontsize=30)
        sub4.set_ylabel('Probability of movement $(t_{2}->t_{3})$', fontsize=30)
        sub4.set_xlabel("$x_{-i}$", fontsize=30)
        sub4.set_xticks(x)
        sub4.set_xticklabels(names, fontsize=20)

        plt.show()
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=5.0), HTML(value='')))
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=5.0), HTML(value='')))
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=5.0), HTML(value='')))
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=5.0), HTML(value='')))
```



12.0.1 Probability of opinion shifts LSL and CSC as functions of x_{-i}, separated by σ_{-i}. Liberals embedded in networks with higher diversity (but a liberal or moderate average opinion) radicalize more frequently.

13 Conditional opinion shift magnitude (only remarkable shifts are considered) (Figure 6)

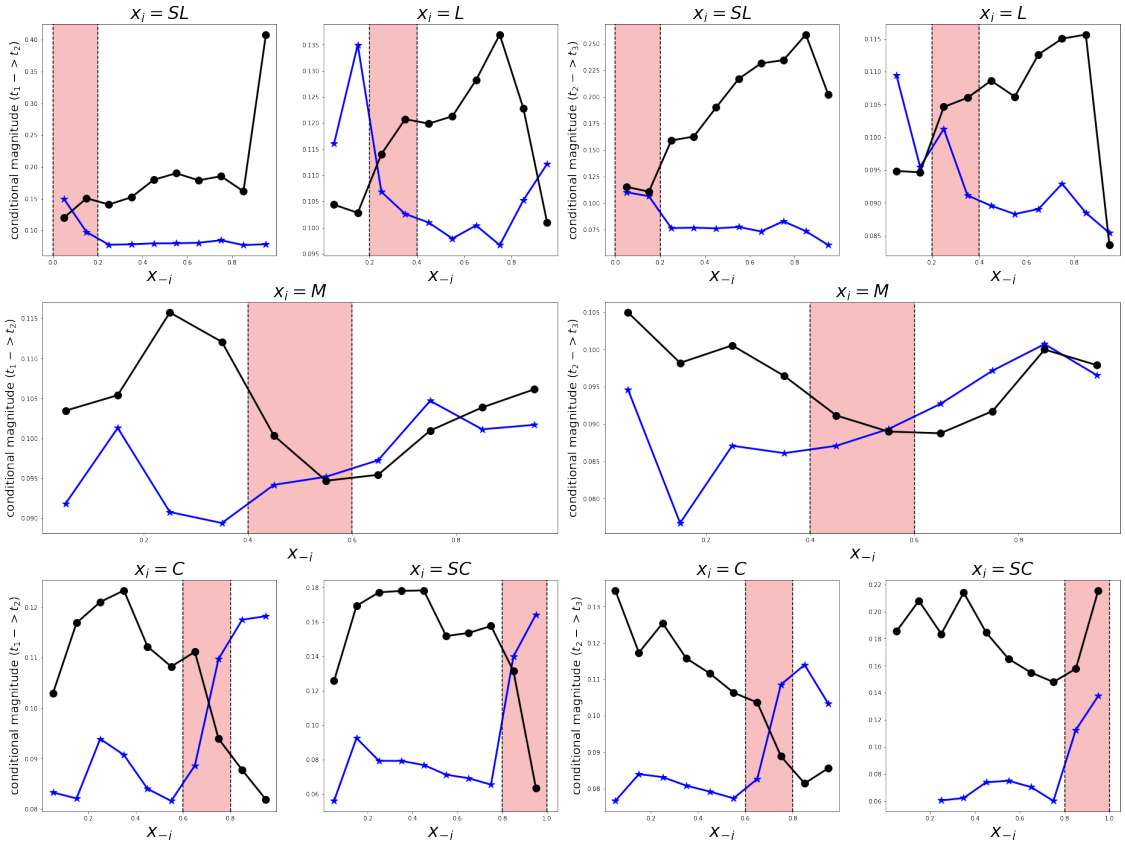
```
In [9]: ths_op = np.arange(0, 1.2, 0.2)
        ths_fr_op = np.arange(0, 1.1, 0.1)

        ##########################################################################

        t1 = 1
```

```
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#Magnitude neg

mask = mask_move & mask_neg & mask_control_fr

table1 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

#Magnitude pos

mask = mask_move & mask_pos & mask_control_fr

table2 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

##############################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, ma

#Magnitude neg

mask = mask_move & mask_neg & mask_control_fr

table3 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

#Magnitude pos

mask = mask_move & mask_pos & mask_control_fr

table4 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)


fig = plt.figure(figsize=(32, 24))


xlabel = "$x_{-i}$"
ylabel1 = 'conditional magnitude $(t_{1} -> t_{2})$'
ylabel2 = 'conditional magnitude $(t_{2} -> t_{3})$'
x = ths_fr_op[1:] - 0.05

########################

sub1 = fig.add_subplot(3, 4, 1) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_magnitude(title_value, k, sub1, x, ths_op, table1, table2)
```

```
########################

sub2 = fig.add_subplot(3, 4, 2) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_magnitude(title_value, k, sub2, x, ths_op, table1, table2)

########################

sub3 = fig.add_subplot(3, 4, 3) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_magnitude(title_value, k, sub3, x, ths_op, table3, table4)

########################

sub4 = fig.add_subplot(3, 4, 4) # 3 rows, 4 columns, 4 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_magnitude(title_value, k, sub4, x, ths_op, table3, table4)

########################

sub5 = fig.add_subplot(3, 4, (5,6))
k = 2
title_value = '$x_{i}=M$'
aux_plot_magnitude(title_value, k, sub5, x, ths_op, table1, table2)

########################

sub6 = fig.add_subplot(3, 4, (7,8))
k = 2
title_value = '$x_{i}=M$'
aux_plot_magnitude(title_value, k, sub6, x, ths_op, table3, table4)

########################

sub7 = fig.add_subplot(3, 4, 9)
k = 3
title_value = '$x_{i}=C$'
aux_plot_magnitude(title_value, k, sub7, x, ths_op, table1, table2)

########################

sub8 = fig.add_subplot(3, 4, 10)
k = 4
title_value = '$x_{i}=SC$'
```

```
aux_plot_magnitude(title_value, k, sub8, x, ths_op, table1, table2)

########################

sub9 = fig.add_subplot(3, 4, 11)
k = 3
title_value = '$x_{i}=C$'
aux_plot_magnitude(title_value, k, sub9, x, ths_op, table3, table4)

########################

sub10 = fig.add_subplot(3, 4, 12)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_magnitude(title_value, k, sub10, x, ths_op, table3, table4)

########################

#sub1.legend(fontsize=15, loc='upper left')
#sub2.legend(fontsize=15, loc='upper left')
#sub3.legend(fontsize=15, loc='upper center')
#sub4.legend(fontsize=15, loc='upper left')
#sub5.legend(fontsize=15, loc='upper right')

size = 30
ysize = 20
#for sub in [sub1, sub2, sub3, sub4, sub5, sub6, sub7, sub8, sub9, sub10]:

#    sub.set_ylim([0, 0.13])

sub1.set_xlabel(xlabel, fontsize=size)
sub2.set_xlabel(xlabel, fontsize=size)
sub3.set_xlabel(xlabel, fontsize=size)
sub4.set_xlabel(xlabel, fontsize=size)
sub5.set_xlabel(xlabel, fontsize=size)
sub6.set_xlabel(xlabel, fontsize=size)
sub7.set_xlabel(xlabel, fontsize=size)
sub8.set_xlabel(xlabel, fontsize=size)
sub9.set_xlabel(xlabel, fontsize=size)
sub10.set_xlabel(xlabel, fontsize=size)

sub1.set_ylabel(ylabel1, fontsize=ysize)
sub3.set_ylabel(ylabel2, fontsize=ysize)
sub5.set_ylabel(ylabel1, fontsize=ysize)
sub6.set_ylabel(ylabel2, fontsize=ysize)
sub7.set_ylabel(ylabel1, fontsize=ysize)
sub9.set_ylabel(ylabel2, fontsize=ysize)
```

```
plt.show()
```



13.0.1 Average opinion shift magnitude as a function of x_{-i} across different values of x_i, separated by movement type. We consider only remarkable shifts. Left panels represent time step t_1t_2; right panels represent time step t_2t_3. Colored areas represent the value of x_i. Black dots plot magnitudes of positive movements; blue starts plot magnitudes of negative movements.

14 Expected opinion shift magnitude (average magnitude of remarkable opinion shift * EPOC)

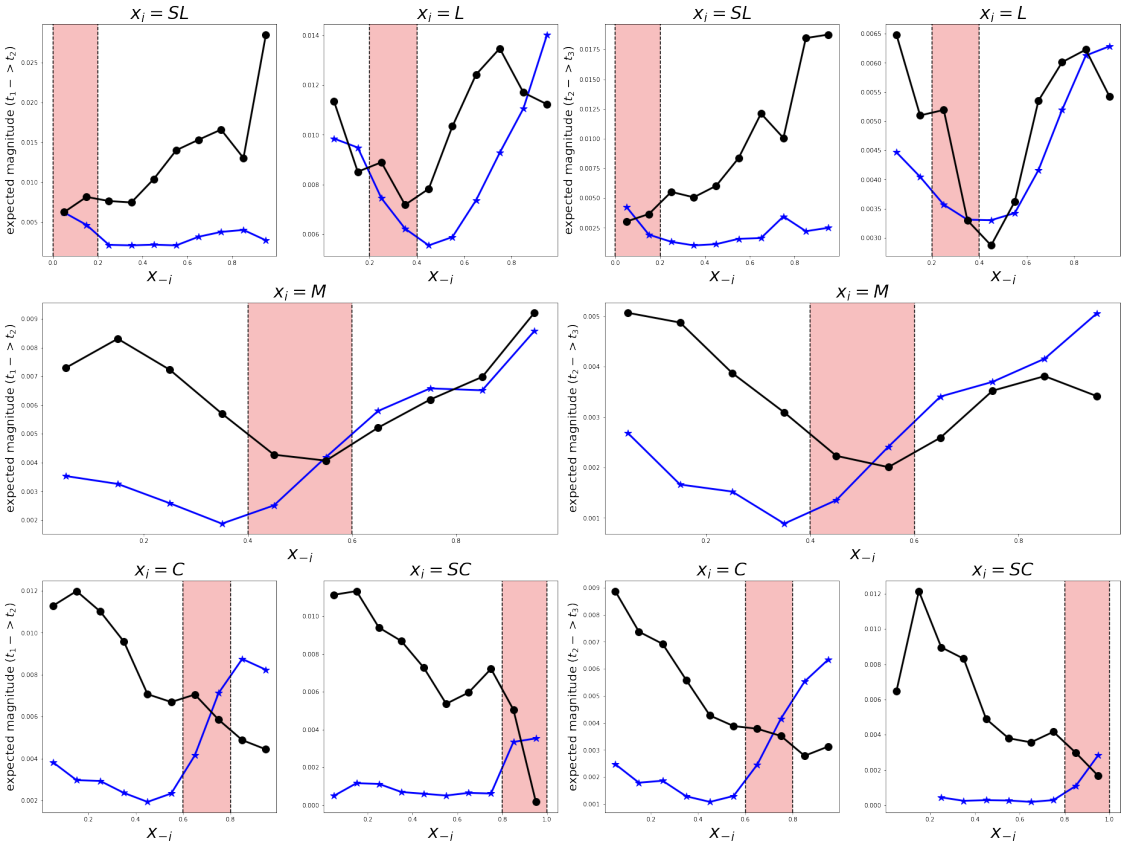
```
In [10]: ths_op = np.arange(0, 1.2, 0.2)
         ths_fr_op = np.arange(0, 1.1, 0.1)

         ##########################################################################

         t1 = 1
         t2 = 2
         op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m
```

```
#EPOC neg

mask_nom = mask_move & mask_neg & mask_control_fr
mask_denom = mask_control_fr

tables1 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_fr
mask_denom = mask_control_fr

tables2 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom

################################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

#EPOC neg

mask_nom = mask_move & mask_neg & mask_control_fr
mask_denom = mask_control_fr

tables3 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom

#EPOC pos

mask_nom = mask_move & mask_pos & mask_control_fr
mask_denom = mask_control_fr

tables4 = fill_tables_EPOC(data, op1, fr_op1, ths_op, ths_fr_op, mask_nom, mask_denom

################################################################################
################################################################################
################################################################################

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

#Magnitude neg

mask = mask_move & mask_neg & mask_control_fr

table1 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)
```

```
#Magnitude pos

mask = mask_move & mask_pos & mask_control_fr

table2 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

##############################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

#Magnitude neg

mask = mask_move & mask_neg & mask_control_fr

table3 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

#Magnitude pos

mask = mask_move & mask_pos & mask_control_fr

table4 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)


table1 = table1*tables1[0]
table2 = table2*tables2[0]
table3 = table3*tables3[0]
table4 = table4*tables4[0]

fig = plt.figure(figsize=(32, 24))


xlabel = "$x_{-i}$"
ylabel1 = 'expected magnitude $(t_{1} -> t_{2})$'
ylabel2 = 'expected magnitude $(t_{2} -> t_{3})$'
x = ths_fr_op[1:] - 0.05

########################

sub1 = fig.add_subplot(3, 4, 1) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_magnitude(title_value, k, sub1, x, ths_op, table1, table2)

########################

sub2 = fig.add_subplot(3, 4, 2) # 3 rows, 4 columns, 2 cell
```

```
k = 1
title_value = '$x_{i}=L$'
aux_plot_magnitude(title_value, k, sub2, x, ths_op, table1, table2)

########################

sub3 = fig.add_subplot(3, 4, 3) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_magnitude(title_value, k, sub3, x, ths_op, table3, table4)

########################

sub4 = fig.add_subplot(3, 4, 4) # 3 rows, 4 columns, 4 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_magnitude(title_value, k, sub4, x, ths_op, table3, table4)

########################

sub5 = fig.add_subplot(3, 4, (5,6))
k = 2
title_value = '$x_{i}=M$'
aux_plot_magnitude(title_value, k, sub5, x, ths_op, table1, table2)

########################

sub6 = fig.add_subplot(3, 4, (7,8))
k = 2
title_value = '$x_{i}=M$'
aux_plot_magnitude(title_value, k, sub6, x, ths_op, table3, table4)

########################

sub7 = fig.add_subplot(3, 4, 9)
k = 3
title_value = '$x_{i}=C$'
aux_plot_magnitude(title_value, k, sub7, x, ths_op, table1, table2)

########################

sub8 = fig.add_subplot(3, 4, 10)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_magnitude(title_value, k, sub8, x, ths_op, table1, table2)

########################
```

```
sub9 = fig.add_subplot(3, 4, 11)
k = 3
title_value = '$x_{i}=C$'
aux_plot_magnitude(title_value, k, sub9, x, ths_op, table3, table4)

########################

sub10 = fig.add_subplot(3, 4, 12)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_magnitude(title_value, k, sub10, x, ths_op, table3, table4)

########################

#sub1.legend(fontsize=15, loc='upper left')
#sub2.legend(fontsize=15, loc='upper left')
#sub3.legend(fontsize=15, loc='upper center')
#sub4.legend(fontsize=15, loc='upper left')
#sub5.legend(fontsize=15, loc='upper right')

size = 30
ysize = 20
#for sub in [sub1, sub2, sub3, sub4, sub5, sub6, sub7, sub8, sub9, sub10]:

#    sub.set_ylim([0, 0.13])

sub1.set_xlabel(xlabel, fontsize=size)
sub2.set_xlabel(xlabel, fontsize=size)
sub3.set_xlabel(xlabel, fontsize=size)
sub4.set_xlabel(xlabel, fontsize=size)
sub5.set_xlabel(xlabel, fontsize=size)
sub6.set_xlabel(xlabel, fontsize=size)
sub7.set_xlabel(xlabel, fontsize=size)
sub8.set_xlabel(xlabel, fontsize=size)
sub9.set_xlabel(xlabel, fontsize=size)
sub10.set_xlabel(xlabel, fontsize=size)

sub1.set_ylabel(ylabel1, fontsize=ysize)
sub3.set_ylabel(ylabel2, fontsize=ysize)
sub5.set_ylabel(ylabel1, fontsize=ysize)
sub6.set_ylabel(ylabel2, fontsize=ysize)
sub7.set_ylabel(ylabel1, fontsize=ysize)
sub9.set_ylabel(ylabel2, fontsize=ysize)

plt.show()
```



15 Control for σ_{-i} (Figure is not included in the main manuscript)

In [13]:
```
ths_op = np.arange(0, 1.2, 0.2)
ths_fr_op = np.arange(0, 1.1, 0.1)

##########################################################################################
##########################################################################################

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

#Magnitude neg

mask = mask_move & mask_neg & (data[fr_op1_divergence] <= 0.15) & mask_control_fr

table1 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

#Magnitude pos
```

```
mask = mask_move & mask_pos & (data[fr_op1_divergence] <= 0.15) & mask_control_fr

table2 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

##############################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

#Magnitude neg

mask = mask_move & mask_neg & (data[fr_op1_divergence] <= 0.15) & mask_control_fr

table3 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

#Magnitude pos

mask = mask_move & mask_pos & (data[fr_op1_divergence] <= 0.15) & mask_control_fr

table4 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

##############################################################################

t1 = 1
t2 = 2
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

#Magnitude neg

mask = mask_move & mask_neg & (data[fr_op1_divergence] > 0.15) & mask_control_fr

table5 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

#Magnitude pos

mask = mask_move & mask_pos & (data[fr_op1_divergence] > 0.15) & mask_control_fr

table6 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

##############################################################################

t1 = 2
t2 = 3
op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

#Magnitude neg
```

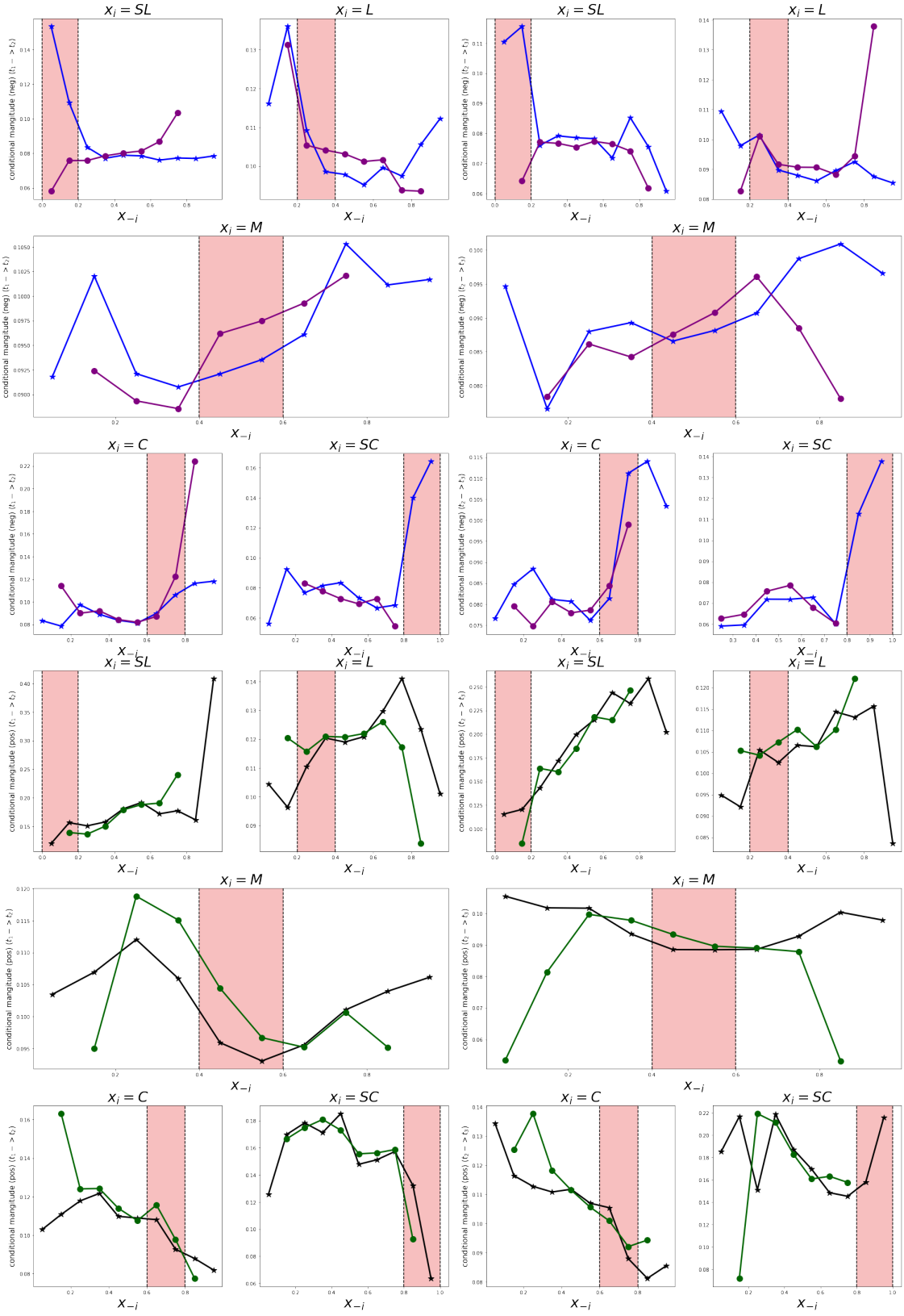
```
mask = mask_move & mask_neg & (data[fr_op1_divergence] > 0.15) & mask_control_fr

table7 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)

#Magnitude pos

mask = mask_move & mask_pos & (data[fr_op1_divergence] > 0.15) & mask_control_fr

table8 = fill_tables_magnitude(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask)


fig = plt.figure(figsize=(32, 48))

#plt.title('A', fontsize=50)
#plt.text(0, 0, 'A', fontsize=50)

xlabel = "$x_{-i}$"
ylabel11 = 'conditional mangitude (neg) $(t_{1} -> t_{2})$'
ylabel12 = 'conditional mangitude (neg) $(t_{2} -> t_{3})$'
ylabel21 = 'conditional mangitude (pos) $(t_{1} -> t_{2})$'
ylabel22 = 'conditional mangitude (pos) $(t_{2} -> t_{3})$'
x = ths_fr_op[1:] - 0.05

########################

sub1 = fig.add_subplot(6, 4, 1) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_magnitude_control_neg(title_value, k, sub1, x, ths_op, table1, table5)

########################

sub2 = fig.add_subplot(6, 4, 2) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_magnitude_control_neg(title_value, k, sub2, x, ths_op, table1, table5)

########################

sub3 = fig.add_subplot(6, 4, 3) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_magnitude_control_neg(title_value, k, sub3, x, ths_op, table3, table7)

########################

sub4 = fig.add_subplot(6, 4, 4) # 3 rows, 4 columns, 4 cell
k = 1
```

```
title_value = '$x_{i}=L$'
aux_plot_magnitude_control_neg(title_value, k, sub4, x, ths_op, table3, table7)

########################

sub5 = fig.add_subplot(6, 4, (5,6))
k = 2
title_value = '$x_{i}=M$'
aux_plot_magnitude_control_neg(title_value, k, sub5, x, ths_op, table1, table5)

########################

sub6 = fig.add_subplot(6, 4, (7,8))
k = 2
title_value = '$x_{i}=M$'
aux_plot_magnitude_control_neg(title_value, k, sub6, x, ths_op, table3, table7)

########################

sub7 = fig.add_subplot(6, 4, 9)
k = 3
title_value = '$x_{i}=C$'
aux_plot_magnitude_control_neg(title_value, k, sub7, x, ths_op, table1, table5)

########################

sub8 = fig.add_subplot(6, 4, 10)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_magnitude_control_neg(title_value, k, sub8, x, ths_op, table1, table5)

########################

sub9 = fig.add_subplot(6, 4, 11)
k = 3
title_value = '$x_{i}=C$'
aux_plot_magnitude_control_neg(title_value, k, sub9, x, ths_op, table3, table7)

########################

sub10 = fig.add_subplot(6, 4, 12)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_magnitude_control_neg(title_value, k, sub10, x, ths_op, table3, table7)

#########################################################################
#########################################################################
```

```
sub11 = fig.add_subplot(6, 4, 13) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub11, x, ths_op, table2, table6)

########################

sub12 = fig.add_subplot(6, 4, 14) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub12, x, ths_op, table2, table6)

########################

sub13 = fig.add_subplot(6, 4, 15) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub13, x, ths_op, table4, table8)

########################

sub14 = fig.add_subplot(6, 4, 16) # 3 rows, 4 columns, 4 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub14, x, ths_op, table4, table8)

########################

sub15 = fig.add_subplot(6, 4, (17,18))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub15, x, ths_op, table2, table6)

########################

sub16 = fig.add_subplot(6, 4, (19,20))
k = 2
title_value = '$x_{i}=M$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub16, x, ths_op, table4, table8)

########################

sub17 = fig.add_subplot(6, 4, 21)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub17, x, ths_op, table2, table6)

########################
```

```
sub18 = fig.add_subplot(6, 4, 22)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub18, x, ths_op, table2, table6)

########################

sub19 = fig.add_subplot(6, 4, 23)
k = 3
title_value = '$x_{i}=C$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub19, x, ths_op, table4, table8)

########################

sub20 = fig.add_subplot(6, 4, 24)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_EPOC_magnitude_pos(title_value, k, sub20, x, ths_op, table4, table8)

#############################################################################
#############################################################################

#sub1.legend(fontsize=15, loc='upper left')
#sub2.legend(fontsize=15, loc='upper left')
#sub3.legend(fontsize=15, loc='upper center')
#sub4.legend(fontsize=15, loc='upper left')
#sub5.legend(fontsize=15, loc='upper right')

size = 15
xsize = 30
#for sub in [sub1, sub2, sub3, sub4, sub5, sub6, sub7, sub8, sub9, sub10]:

#    sub.set_ylim([0, 0.13])

sub1.set_xlabel(xlabel, fontsize=xsize)
sub2.set_xlabel(xlabel, fontsize=xsize)
sub3.set_xlabel(xlabel, fontsize=xsize)
sub4.set_xlabel(xlabel, fontsize=xsize)
sub5.set_xlabel(xlabel, fontsize=xsize)
sub6.set_xlabel(xlabel, fontsize=xsize)
sub7.set_xlabel(xlabel, fontsize=xsize)
sub8.set_xlabel(xlabel, fontsize=xsize)
sub9.set_xlabel(xlabel, fontsize=xsize)
sub10.set_xlabel(xlabel, fontsize=xsize)

sub1.set_ylabel(ylabel11, fontsize=size)
sub3.set_ylabel(ylabel12, fontsize=size)
```

```
sub5.set_ylabel(ylabel11, fontsize=size)
sub6.set_ylabel(ylabel12, fontsize=size)
sub7.set_ylabel(ylabel11, fontsize=size)
sub9.set_ylabel(ylabel12, fontsize=size)

sub11.set_xlabel(xlabel, fontsize=xsize)
sub12.set_xlabel(xlabel, fontsize=xsize)
sub13.set_xlabel(xlabel, fontsize=xsize)
sub14.set_xlabel(xlabel, fontsize=xsize)
sub15.set_xlabel(xlabel, fontsize=xsize)
sub16.set_xlabel(xlabel, fontsize=xsize)
sub17.set_xlabel(xlabel, fontsize=xsize)
sub18.set_xlabel(xlabel, fontsize=xsize)
sub19.set_xlabel(xlabel, fontsize=xsize)
sub20.set_xlabel(xlabel, fontsize=xsize)

sub11.set_ylabel(ylabel21, fontsize=size)
sub13.set_ylabel(ylabel22, fontsize=size)
sub15.set_ylabel(ylabel21, fontsize=size)
sub16.set_ylabel(ylabel22, fontsize=size)
sub17.set_ylabel(ylabel21, fontsize=size)
sub19.set_ylabel(ylabel22, fontsize=size)

plt.show()
```


$x_i = SL$
$x_i = L$
$x_i = SL$
$x_i = L$
$x_i = M$
$x_i = M$
$x_i = C$
$x_i = SC$
$x_i = C$
$x_i = SC$
$x_i = SL$
$x_i = L$
$x_i = SL$
$x_i = L$
$x_i = M$
$x_i = M$
$x_i = C$
$x_i = SC$
$x_i = C$
$x_i = SC$
x_{-i}

15.0.1 In this plot, we demonstrate how conditional opinion shift magnitude is moderated by the value of σ_{-i}. Blue stars plot opinion shift magnitude if σ_{-i} is under 0.15; purple dots plot opinion shift magnitude if $\sigma_{-i}0.15$. Black stars plot opinion shift magnitude if σ_{-i} is under 0.15; green dots plot opinion shift magnitude if $\sigma_{-i}0.05$. We observe no clear trends.

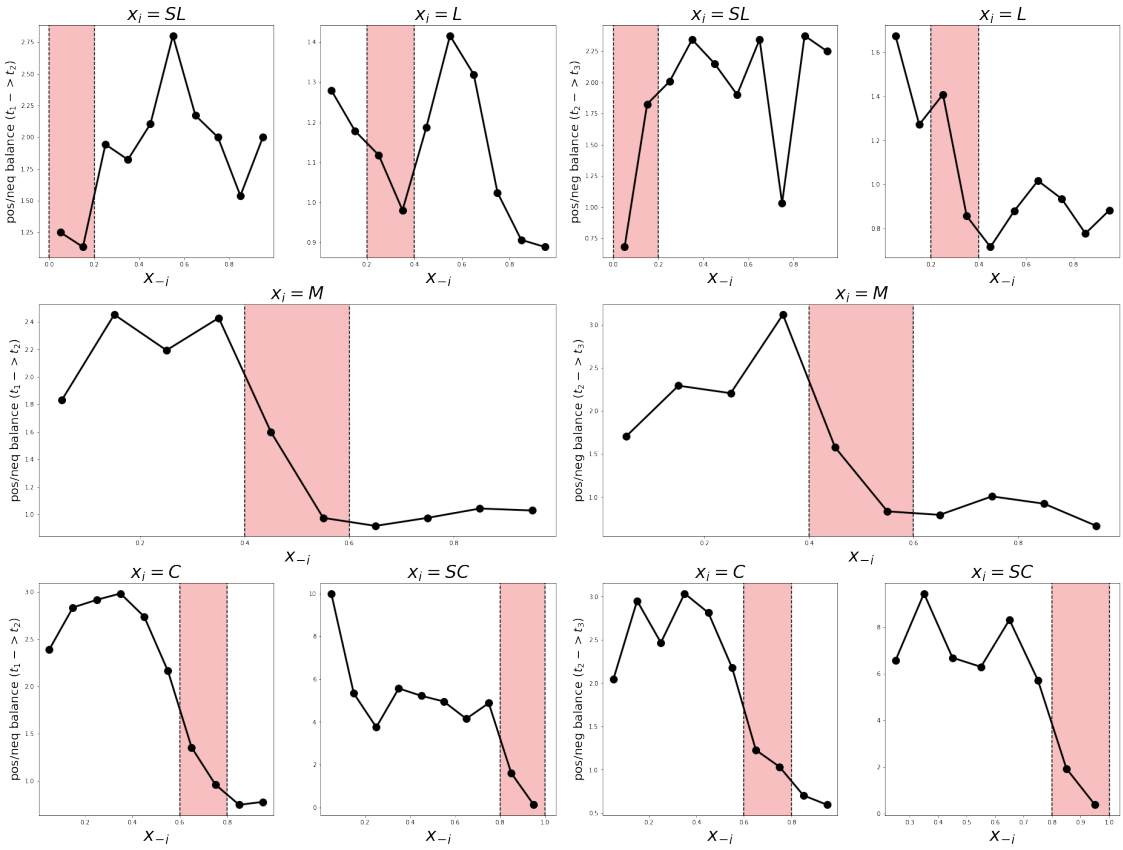
16 The positive / negative ratio as a function of x_{-i}, control for x_i (Figure 7)

```
In [15]: ths_op = np.arange(0, 1.2, 0.2) #[0, 0.35, 0.45, 0.55, 0.65, 1]
         ths_fr_op = np.arange(0, 1.1, 0.1) #[0, 0.35, 0.45, 0.55, 0.65, 1]

         ##############################################################################################

         t1 = 1
         t2 = 2
         op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

         mask_nom = mask_move & mask_pos & mask_control_fr
         mask_denom = mask_move & mask_neg & mask_control_fr



         table1 = fill_tables_balance(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask_nom

         ##############################################################################################

         t1 = 2
         t2 = 3
         op1, op2, fr_op1, fr_op2, fr_op1_divergence, op_change, anchor, mask_move, mask_neg, m

         mask_nom = mask_move & mask_pos & mask_control_fr
         mask_denom = mask_move & mask_neg & mask_control_fr



         table2 = fill_tables_balance(data, op_change, op1, fr_op1, ths_op, ths_fr_op, mask_nom

         ##############################################################################################


         fig = plt.figure(figsize=(32, 24))


         xlabel = "$x_{-i}$"
         ylabel1 = 'pos/neq balance $(t_{1} -> t_{2})$'
```

```
ylabel2 = 'pos/neg balance $(t_{2} -> t_{3})$'
x = ths_fr_op[1:] - 0.05

########################

sub1 = fig.add_subplot(3, 4, 1) # 3 rows, 4 columns, 1 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_balance(title_value, k, sub1, x, ths_op, table1)

########################

sub2 = fig.add_subplot(3, 4, 2) # 3 rows, 4 columns, 2 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_balance(title_value, k, sub2, x, ths_op, table1)

########################

sub3 = fig.add_subplot(3, 4, 3) # 3 rows, 4 columns, 3 cell
k = 0
title_value = '$x_{i}=SL$'
aux_plot_balance(title_value, k, sub3, x, ths_op, table2)

########################

sub4 = fig.add_subplot(3, 4, 4) # 3 rows, 4 columns, 4 cell
k = 1
title_value = '$x_{i}=L$'
aux_plot_balance(title_value, k, sub4, x, ths_op, table2)

########################

sub5 = fig.add_subplot(3, 4, (5,6))
k = 2
title_value = '$x_{i}=M$'
aux_plot_balance(title_value, k, sub5, x, ths_op, table1)

########################

sub6 = fig.add_subplot(3, 4, (7,8))
k = 2
title_value = '$x_{i}=M$'
aux_plot_balance(title_value, k, sub6, x, ths_op, table2)

########################

sub7 = fig.add_subplot(3, 4, 9)
```

```
k = 3
title_value = '$x_{i}=C$'
aux_plot_balance(title_value, k, sub7, x, ths_op, table1)

########################

sub8 = fig.add_subplot(3, 4, 10)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_balance(title_value, k, sub8, x, ths_op, table1)

########################

sub9 = fig.add_subplot(3, 4, 11)
k = 3
title_value = '$x_{i}=C$'
aux_plot_balance(title_value, k, sub9, x, ths_op, table2)

########################

sub10 = fig.add_subplot(3, 4, 12)
k = 4
title_value = '$x_{i}=SC$'
aux_plot_balance(title_value, k, sub10, x, ths_op, table2)

########################

#sub1.legend(fontsize=15, loc='upper left')
#sub2.legend(fontsize=15, loc='upper left')
#sub3.legend(fontsize=15, loc='upper center')
#sub4.legend(fontsize=15, loc='upper left')
#sub5.legend(fontsize=15, loc='upper right')

size = 30
ysize = 20
#for sub in [sub1, sub2, sub3, sub4, sub5, sub6, sub7, sub8, sub9, sub10]:

#    sub.set_ylim([0, 0.13])

sub1.set_xlabel(xlabel, fontsize=size)
sub2.set_xlabel(xlabel, fontsize=size)
sub3.set_xlabel(xlabel, fontsize=size)
sub4.set_xlabel(xlabel, fontsize=size)
sub5.set_xlabel(xlabel, fontsize=size)
sub6.set_xlabel(xlabel, fontsize=size)
sub7.set_xlabel(xlabel, fontsize=size)
sub8.set_xlabel(xlabel, fontsize=size)
sub9.set_xlabel(xlabel, fontsize=size)
```

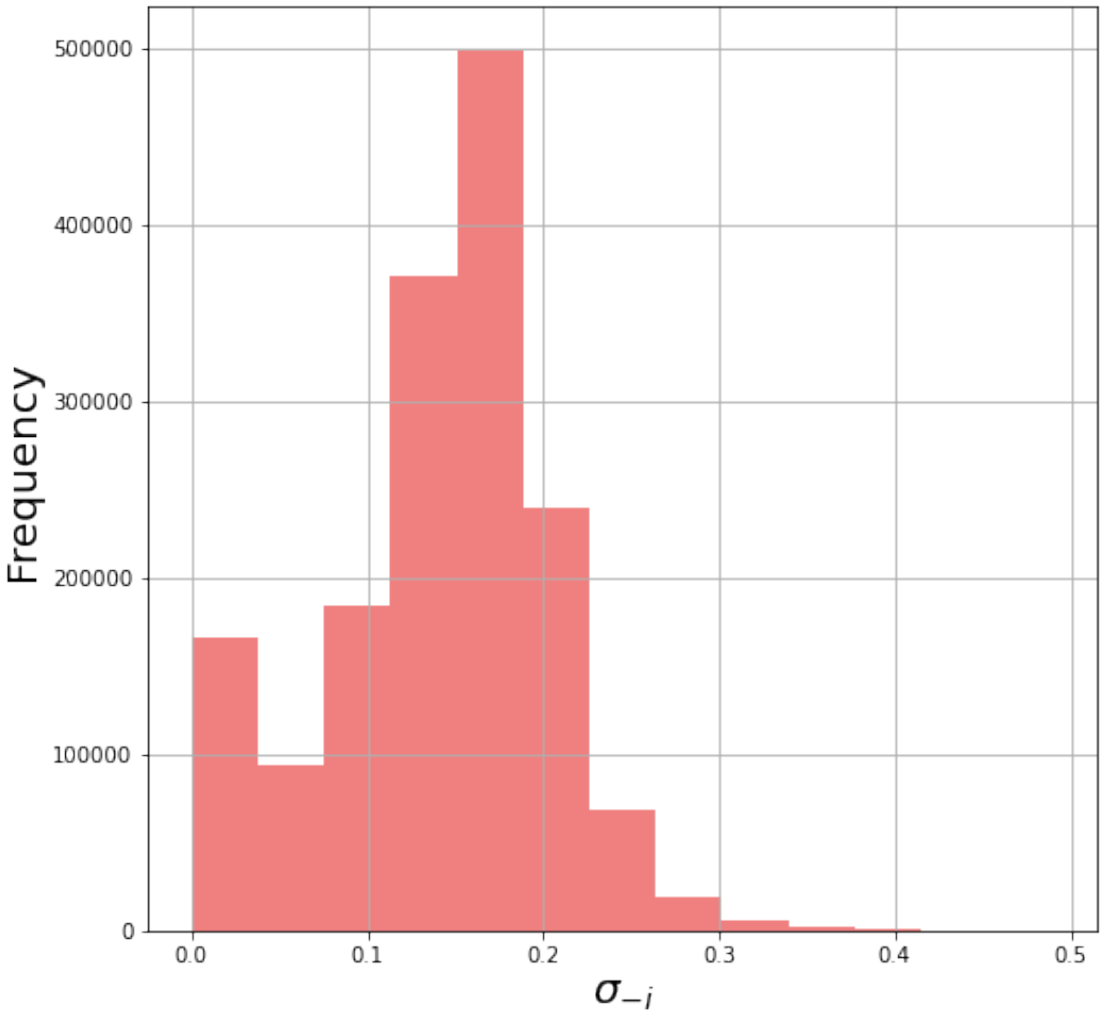
```
sub10.set_xlabel(xlabel, fontsize=size)

sub1.set_ylabel(ylabel1, fontsize=ysize)
sub3.set_ylabel(ylabel2, fontsize=ysize)
sub5.set_ylabel(ylabel1, fontsize=ysize)
sub6.set_ylabel(ylabel2, fontsize=ysize)
sub7.set_ylabel(ylabel1, fontsize=ysize)
sub9.set_ylabel(ylabel2, fontsize=ysize)

plt.show()
```



16.0.1 Positive/negative ratio as a function of x_{-i}, separated by x_i. Left panels represent time step t_1t_2; right panels represent time step t_2t_3. Colored areas represent the value of x_i.

17 Histogram of σ_{-i}

In [77]:
```
plt.figure(figsize=(8, 8))
data[fr_op1_divergence].hist(bins=13, color='lightcoral')
plt.xlabel('$\sigma_{-i}$', fontsize=20)
plt.ylabel('Frequency', fontsize=20)
plt.show()
```


500000
400000
300000
200000
100000
0
Frequency
0.0
0.1
0.2
0.3
0.4
0.5
σ_{-i}