# Each dataset corresponds to a set of figures. To generate each of the figures, use the following Matlab Scripts along with the relevant datasets:

# Figure 4 - Validation Data.xlsx

# Figure 5 - Validation Data.xlsx

# Figure 6 - Predicted flow times\_scatter data.xlsx

# Figure 7 -

# Ensure the dataset files are contained in the same file as the Matlab script location.

-------------------------

# Figure 4

# Read the data from the Validation Data.xlsx file. The script reads the entire data set for each station location.

# Plot the bar plots using the following Matlab script.

Validation\_Data\_Birmingham = xlsread('Validation Data','Birmingham New Street','C2:D6');

Validation\_Data\_Leeds = xlsread('Validation Data','Leeds','C2:D18');

Validation\_Data\_LDN = xlsread('Validation Data','London Bridge','C2:D4');

subplot(2,2,[1,2])

bar(Validation\_Data\_Leeds);

xticks([1:1:17])

xticklabels({'1','2','3','4','5','6','7','8','9','10','11','12','13','14','15','16','17'});

xlabel({'Case number','Leeds'});

ylabel('Flow time (s)');

legend('Predicted','Measured','Location','northwest');

legend boxoff

subplot(2,2,3)

bar(Validation\_Data\_Birmingham);

xticks([1:1:5])

xticklabels({'18','19','20','21','5','22'});

xlabel({'Case number','Birmingham New Street'});

ylabel('Flow time (s)');

subplot(2,2,4)

bar(Validation\_Data\_LDN);

xticks([1:1:3])

xticklabels({'23','24','25'});

xlabel({'Case number','London Bridge'});

ylabel('Flow time (s)');

-------------------------

# Figure 5

Validation\_Data\_Birmingham = xlsread('Validation Data','Birmingham New Street','C2:D6');

Validation\_Data\_Leeds = xlsread('Validation Data','Leeds','C2:D18');

Validation\_Data\_LDN = xlsread('Validation Data','London Bridge','C2:D4');

hold on

plot(Validation\_Data\_Birmingham(:,1),Validation\_Data\_Birmingham(:,2),'bx')

plot([0:1:80],[0:1:80],'--');

plot([0:1:80],1.044\*[0:1:80]);

plot(Validation\_Data\_Leeds(:,1),Validation\_Data\_Leeds(:,2),'bx')

plot(Validation\_Data\_LDN(:,1),Validation\_Data\_LDN(:,2),'bx')

hold off

xlabel('Predicted flow time (s)');

ylabel('Measured flow time (s)');

legend('Flow time','1:1 ideal case','Linear fit')

-------------------------

# The data for the scatter plots in Figure 6 is contained within 'Predicted flow times\_scatter data.xlsx'. Each train and platform combination is contained with the relevant sheet.

# Figures 6(a)--(d)

# Read the required dataset from the relevant sheet name. Each column corresponds to the level of social distancing considered.

Scatter\_Data = xlsread('Predicted flow times\_scatter data','Sheet Name');

hold on

plot(Scatter\_Data(:,1),Scatter\_Data(:,2),'.')

plot(Scatter\_Data(:,1),Scatter\_Data(:,3),'x')

plot(Scatter\_Data(:,1),Scatter\_Data(:,4),'^')

hold off

xlabel('Total number of passengers per door');

ylabel('Predicted flow time (s)');

legend('0 m SD',’0.75 m’,'1 m SD','2m SD');

-------------------------

# The plane fit used to generate Figure 7 can be obtained with the following Matlab script. The coefficients for the fitted planes obtained using this method are in Table 4 for the other modelled cases.

Pax = xlsread('Ratesetter Data\_Intercity 6 m platform','1 m SD','C2:J2');

Time = xlsread('Ratesetter Data\_Intercity 6 m platform','1 m SD','C3:J10');

[xData, yData, zData] = prepareSurfaceData( Pax, Pax, Time );

% Set up fittype and options.

ft = fittype( 'poly11' );

% Fit model to data.

[fitresult, gof] = fit( [xData, yData], zData, ft );

% Plot fit with data.

figure( 'Name', 'Linear plane fit to predicted flow times' );

h = plot( fitresult, [xData, yData], zData );

legend( h, 'Linear plane fit $f(x,y)$', 'Predicted flow times', 'Location', 'NorthEast', 'Interpreter', 'latex' );

% Label axes

xlabel( 'Boarding passengers per door', 'Interpreter', 'none' );

ylabel( 'Alighting passengers per door', 'Interpreter', 'none' );

zlabel( 'Predicted flow times', 'Interpreter', 'none' );

grid off