```
%% ----- Exercise 5 - Naturalistic data analysis -----
% Version: 2022
% Course: TME 192 Active Safety
%         Chalmers
% Author: Alberto Morando (morando@chalmers.se)
%         Alexander Rasch (arasch@chalmers.se)
%         Pierluigi Olleja (ollejap@chalmers.se)
%         Marco Dozza (dozza@chalmers.se)
%
%  Group number: [Group 5]
%  Group member: [Yahui Wu]
%  Group member: [Tianshuo Xiao]
%  Group member: [Nishanth Suresh]

close all
clear all
clc

%% Initialization
% Colormap for plotting
cmap = [214,96,77;
    67,147,195]./255;

% to keep track of discarded events
num_discarded_crash_events = 0;
num_discarded_near_crash_events = 0;

% to keep track of the valid event IDs
crash_event_ID = [];
near_crash_event_ID = [];

% these arrays contain the mean speed for each event
crash_mean_speed = [];
near_crash_mean_speed = [];

%% Query data
% Collect the .mat event filenames in the "Events" folder. Here you can use the  ?
dir? function
EventFolder = 'Events';
EventFiles = dir(fullfile(EventFolder, '*.mat'));

% Loop throught the event files.
for i = 1:length(EventFiles)

    % Load single events data. Here you can use the  ?load? function
    load(fullfile(EventFolder, EventFiles(i).name));

    % ======== YOUR CODE HERE ========

    % Analyse the event only if the incident type is 'Rear-end striking'. Look at
    % the dictionary "ResearcherDictionaryVideoReductionDatav1_1" to understand how
to extract
    % this information. Here you can use the ?strcmpi? function
    if strcmpi('Rear-end striking',Data100Car.Video.incident_type)

        % Extract the ?speed? data. Have a look at the dictionary
"100CarTimeSeriesDataDictionary_v1_1" to
        % understand how to extract this information
        speed = Data100Car.TimeSeries(:,5);
        % Check speed quality with the function "isGoodQuality".
        % If the quality is not good we will skip the event.
        if isGoodQuality(speed)

            % Extract time [s]. Make it start from 0
            time = Data100Car.TimeSeries(:,3) - Data100Car.TimeSeries(1,3);

            % The speed signal may be corrupted (missing data or sensor
            % error). Here we will implement a simple, yet effective, strategy for
dealing
            % with this issue. We will perform a linear interpolation to
            % recover the missing data. Specifically, interpolate the
            % values that are less or equal to zero.

            % Make a copy of the vector
            speed_fix = speed;

            % Interpolate the speed
            speed_fix(speed_fix<=0) =
interp1(time(speed_fix>0),speed_fix(speed_fix>0),time(speed_fix<=0),'linear','extra
p');

            % extract the speed only during the incident. Yout could use `find`
            idx_start_incident = find(Data100Car.TimeSeries(:,2) ==
Data100Car.Video.start);
            idx_end_incident = find(Data100Car.TimeSeries(:,2) ==
Data100Car.Video.end);
            speed_incident = speed_fix(idx_start_incident:idx_end_incident);

            % Check if the event was a `Crash` or `Near Crash`. Depending
            % on the incident category, store the event ID and the mean
            % speed in the right variable.
            % Take a look at the dictionary
"ResearcherDictionaryVideoReductionDatav1_1" to understand how to extract
            % this information.
            if  strcmpi('Crash',Data100Car.Video.severity)% If `Crash`
                % save the ID of the event and append it to the vector
                crash_event_ID = [crash_event_ID;Data100Car.ID];
                % compute and append average speed to the vector
                crash_mean_speed = [crash_mean_speed; mean(speed_incident)];

            elseif strcmpi('Near Crash',Data100Car.Video.severity) % If `Near
Crash`
                % save the ID of the event and append it to the vector
                near_crash_event_ID = [near_crash_event_ID; Data100Car.ID];
                % compute and append average speed to the vector
                near_crash_mean_speed = [near_crash_mean_speed;
mean(speed_incident)];
            end

        else
            % If the data quality is low, we discard the event. However, we
            % will keep track of how many events were discarded per category.
            if strcmpi('Crash',Data100Car.Video.severity) % If `Crash`
                num_discarded_crash_events = num_discarded_crash_events+1;
            elseif strcmpi('Near Crash',Data100Car.Video.severity) % If `Near
Crash`
                num_discarded_near_crash_events = num_discarded_near_crash_events+1
```

```
;
            end
        end
    end
end


%% Proportion [%] of missing data.
% ======== YOUR CODE HERE ========
% from the number of discarded events and the number of valid ones, compute the
percentage of
% missing data for the crash and near-crash events
percentageMissingDataCrashes =
num_discarded_crash_events/(num_discarded_crash_events+length(crash_event_ID));
percentageMissingDataNearCrashes =
num_discarded_near_crash_events/(num_discarded_near_crash_events+length(near_crash_
event_ID));

fprintf('Percentage missing data for crash events = %3.2f %%\n',
percentageMissingDataCrashes)
fprintf('Percentage missing data for near-crash events = %3.2f %%\n',
percentageMissingDataNearCrashes)

%% QUESTION
% Is the percentage of missing data you found usual for a naturalistic dataset?
% (Enclose your answer in a comment)
% From the percentages above it can be concluded that the missing data rate is
% very low (0.29% missing crash events and 0.12% missing near-crash events).
% High-quality datasets usually have low missing data rates, and after processing
% our data, we have a more common percentage of missing data for natural datasets.

%% Statistics: compare the samples. Is the average speed is significantly lower in
crashes than in near crashes?
% Compare the speed for Near-crashes and crashes with a t-test.
% Before running the t-test, you should verify that the data meet the
% assumption of normality. If the data are not normally distribuited, the result
from the t-test may be misleading.
% To do so, plot the average speed values for the crash and near-crash
% events with a histogram (use function ?histogram?).
bins = 0:5:100;
x_pdf = [0:0.1:max(bins)];

figure('name', 'original data')
hold on

% Distribution mean speed for near-crashes
histogram(near_crash_mean_speed, bins,...
    'normalization', 'pdf', ...
    'facecolor', cmap(1, :))

% Fit a normal distribution to the data
y = pdf(fitdist(near_crash_mean_speed,'Normal'), x_pdf);
plot(x_pdf, y, '-', 'color', cmap(1, :))

% Distribution mean speed for near-crashes
histogram(crash_mean_speed, bins,...
    'normalization', 'pdf',...
    'facecolor', cmap(2, :))
```

```
% Fit a normal distribution to the data
y = pdf(fitdist(crash_mean_speed,'Normal'), x_pdf);
plot(x_pdf, y, '-', 'color', cmap(2, :))

xlabel('Mean speed [mph]')
ylabel('Probability density')

legend({'Near crashes', 'Fitted Normal distribution Near Crashes',...
    'Crashes', 'Fitted Normal distribution Crashes'})

%% Data transformation
% If a measurement variable does not quite fit a normal distribution, you should
try a data transformation.
% For example, you could try to apply the square root (`sqrt`). Does the
% transformation make the data fit the normality assumption better?

% ======== YOUR CODE HERE ========
near_crash_mean_speed_tranformed = sqrt(near_crash_mean_speed);
crash_mean_speed_tranformed = sqrt(crash_mean_speed);

bins = 0:0.5:10;
x_pdf = [0:0.1:max(bins)];

figure('name', 'trasnformed data')
hold on

% Distribution mean speed for near-crashes
histogram(near_crash_mean_speed_tranformed, bins,...
    'normalization', 'pdf', ...
    'facecolor', cmap(1, :))

% Fit a normal distribution to the data
y = pdf(fitdist(near_crash_mean_speed_tranformed,'Normal'), x_pdf);
plot(x_pdf, y, '-', 'color', cmap(1, :))

% Distribution mean speed for near-crashes
histogram(crash_mean_speed_tranformed, bins,...
    'normalization', 'pdf',...
    'facecolor', cmap(2, :))

% Fit a normal distribution to the data
y = pdf(fitdist(crash_mean_speed_tranformed,'Normal'), x_pdf);
plot(x_pdf, y, '-', 'color', cmap(2, :))

xlabel('Mean speed [mph]')
ylabel('Probability density')

legend({'Near crashes', 'Fitted Normal distribution Near Crashes', ...
    'Crashes', 'Fitted Normal distribution Crashes'})

%% t-test
% Hyhothesis to test ?Average velocity is higher in rear-end striking near-crashes
than in
% rear-end striking crashes?. Run the ttest. Use the function ?ttest2?. You should
set the option `tail` according to the hyphothesis
% you are testing. Be careful on the order of the data you input.

% ======== YOUR CODE HERE ========
[H,P, ~, ~] = ttest2(near_crash_mean_speed_tranformed,crash_mean_speed_tranformed,
'tail', 'right', 'Vartype','unequal')

%% QUESTION
% Does the ttest support the hyp. that ?Average velocity is higher in rear-end
striking near-crashes than in rear-end striking crashes??
% (Enclose your answer in a comment)
% In our results H=1, indicates that there is a significant difference between
% the average velocity in rear-end striking near-crashes and in rear-end striking
crashes
% for both events. The P value is 6.4966e-05 < 0.05, so there is statistical
evidence to support
% the hypothesis that the average velocity in rear-end striking near-crashes is
higher than the
% average velocity in rear-end striking crashes.

%% Explore the narrative of the valid crash events
% The list of event IDs to load are in the vector ?crash_event_ID?.

% Loop over the event files. Like we did in the beginning of this script
for i = 1:length(EventFiles)

    % ======== YOUR CODE HERE ========
    % Load single events data. Here you can use the  ?load? function
    load(fullfile(EventFolder, EventFiles(i).name));

    % Check if the event ID is included in the vector `crash_event_ID`. You
    % can use `ismember`
    if ismember(Data100Car.ID,crash_event_ID)

        % Print the narrative on the console.
        narrative = Data100Car.Narratives{1,1};
        fprintf('Event %d: %s\n', Data100Car.ID, narrative)
    end
end

%% QUESTION
% By reading the narrative of the event, what do you think is
% the most common scenario and critical reason that lead to a crash?
% (Enclose your answer in a comment)
% In these events, inattention or distraction appears to be a common cause of
crashes .
% The main cause of these crashes appears to be the inability of drivers to
% maintain their focus on the road and react quickly to changing traffic
conditions.
```