%% PREDICTORS: SPEED CONDITION, RESPONSE: KINEMATICS, STATS TEST

STATS\_OUT = [];

im\_resize= 1.2;

VIOLIN\_BOTTOM = 0.7;

AX\_H = 0.2;

AX\_W = 0.25;

DO\_PLOT\_GROUPS = false;

tmp\_savedir = [save\_dir filesep 'Pspeed-Rkin'];

mkdir(tmp\_savedir);

for var\_i = 1:length(varnames)

%

vert\_shift = 0;

for des\_i = 2 %## JUST SPEED

%##

horiz\_shift = 0;

switch des\_i

case 1

color\_dark = COLORS\_MAPS\_TERRAIN;

color\_light = COLORS\_MAPS\_TERRAIN;

GROUP\_CMAP\_OFFSET = [0,0.1,0.1];

xtick\_label\_g = {'flat','low','med','high'};

case 2

color\_dark = COLOR\_MAPS\_SPEED;

color\_light = COLOR\_MAPS\_SPEED+0.15;

GROUP\_CMAP\_OFFSET = [0.15,0,0];

xtick\_label\_g = {'0.25','0.50','0.75','1.0'};

end

inds = TMP\_FOOOF\_T.design\_id == designs(des\_i);

T\_vals\_plot = TMP\_FOOOF\_T(inds,:);

subjects = unique(T\_vals\_plot.subj\_char);

conds = unique(T\_vals\_plot.cond\_id);

% groups = unique(T\_vals\_plot.group\_id);

t\_tmp = [];

for i = 1:length(subjects)

ii = find(T\_vals\_plot.subj\_char == subjects(i));

tt = T\_vals\_plot(ii,:);

for j = 1:length(conds)

jj = find(tt.cond\_id == conds(j));

t\_tmp = [t\_tmp; tt(jj(1),:)];

end

end

T\_vals\_plot = table(categorical(string(t\_tmp.cond\_char)),t\_tmp.(varnames{var\_i}),categorical(string(t\_tmp.group\_char)),...

'VariableNames',{'cond\_char',varnames{var\_i},'group\_char'});

% T\_vals\_plot.cond\_char = double(string(T\_vals\_plot.cond\_char));

try

mod = sprintf('%s ~ 1 + %s',varnames{var\_i},'cond\_char');

% stats\_out = fitlme(T\_vals\_plot,mod);

stats\_out = fitlm(T\_vals\_plot,mod);

% anova\_out = anova(stats\_out);

out = anova(T\_vals\_plot,mod,'SumOfSquaresType',"three",'CategoricalFactors',{'cond\_char'},...

'ModelSpecification','linear');

anova\_out = out.stats();

% anova\_out = anovan(double(T\_vals\_plot.(varnames{var\_i})),{T\_vals\_plot.cond\_char},...

% 'model','linear',...

% 'model',1,...

% 'sstype',3,...

% 'varnames',strvcat('speed'));

%## PRINT TABLES

disp(anova\_out);

disp(stats\_out);

% t = sprintf\_table(anova\_out);

% t.print;

% t.saveToFile([tmp\_savedir filesep sprintf('%s\_kinematics-speed\_ANOVA.tex',varnames{var\_i})]);

% t = sprintf\_table(stats\_out.Coefficients);

% t.print;

% t.saveToFile([tmp\_savedir filesep sprintf('%s\_kinematics-speed\_LM.tex',varnames{var\_i})]);

%-

R2 = stats\_out.Rsquared.Adjusted;

anova\_p\_var = anova\_out.pValue(strcmp(anova\_out.Properties.RowNames,'cond\_char'));

pval\_inter = double(stats\_out.Coefficients.pValue(strcmp(stats\_out.Coefficients.Properties.RowNames,'(Intercept)')));

pval\_var\_0p5 = stats\_out.Coefficients.pValue(strcmp(stats\_out.Coefficients.Properties.RowNames,'cond\_char\_0.5'));

pval\_var\_0p75 = stats\_out.Coefficients.pValue(strcmp(stats\_out.Coefficients.Properties.RowNames,'cond\_char\_0.75'));

pval\_var\_1p0 = stats\_out.Coefficients.pValue(strcmp(stats\_out.Coefficients.Properties.RowNames,'cond\_char\_1.0'));

% tstat\_var = stats\_out.Coefficients.tStat(strcmp(stats\_out.Coefficients.Properties.RowNames,'cond\_char'));

% slope\_var = double(stats\_out.Coefficients.Estimate(strcmp(stats\_out.Coefficients.Properties.RowNames,'cond\_char')));

inter\_mn = double(stats\_out.Coefficients.Estimate(strcmp(stats\_out.Coefficients.Properties.RowNames,'(Intercept)')));

catch e

fprintf('Error. Cluster %s\n\n%s\n',string(clusters(cl\_i)),getReport(e))

R2 = 0;

pval = 1;

slope = 0;

inter = 0;

end

%##

STATS\_STRUCT = struct('anova',{{}},...

'anova\_grp',{{}},...

'pvals',{{}},...

'pvals\_pairs',{{}},...

'pvals\_grp',{{}},...

'pvals\_grp\_pairs',{{}},...

'regress\_pval',{{}},...

'regress\_line',{{}},...

'r2\_coeff',{{}},...

'regress\_xvals',0);

if DO\_PLOT\_GROUPS

for gg = 1:length(groups)

STATS\_STRUCT.anova{gg}=anova\_p\_var;

STATS\_STRUCT.pvals\_pairs{gg}={[1,1],[1,2],[1,3],[1,4]};

STATS\_STRUCT.pvals{gg}=[pval\_inter,pval\_var\_0p5,pval\_var\_0p75,pval\_var\_1p0];

end

else

STATS\_STRUCT.anova{1}=anova\_p\_var;

STATS\_STRUCT.pvals\_pairs{1}={[1,1],[1,2],[1,3],[1,4]};

STATS\_STRUCT.pvals{1}=[pval\_inter,pval\_var\_0p5,pval\_var\_0p75,pval\_var\_1p0];

end

STATS\_OUT = [STATS\_OUT; STATS\_STRUCT];

% figure;

VIOLIN\_PARAMS = {'width',0.1,...

'ShowWhiskers',false,'ShowNotches',false,'ShowBox',true,...

'ShowMedian',true,'Bandwidth',0.15,'QuartileStyle','shadow',...

'HalfViolin','full','DataStyle','scatter','MarkerSize',8,...

'EdgeColor',[0.5,0.5,0.5],'ViolinAlpha',{0.2 0.3}};

PLOT\_PARAMS = struct('color\_map',color\_dark,...

'cond\_labels',unique(T\_vals\_plot.cond\_char),'group\_labels',unique(T\_vals\_plot.group\_char),...

'cond\_offsets',[-0.35,-0.1,0.15,0.4],'y\_label',varnames\_labs{var\_i},...

'title',varnames\_labs{var\_i},'font\_size',10,'group\_offsets',[0.125,0.475,0.812],...

'ylim',[min(T\_vals\_plot.(varnames{var\_i}))-std(T\_vals\_plot.(varnames{var\_i})),max(T\_vals\_plot.(varnames{var\_i}))+3\*std(T\_vals\_plot.(varnames{var\_i}))],...

'font\_name','Arial','x\_label','speed','do\_combine\_groups',~DO\_PLOT\_GROUPS);

fig = figure('color','white','renderer','Painters');

set(fig,'Units','inches','Position',[0.5,0.5,3,3])

set(fig,'PaperUnits','inches','PaperSize',[1 1],'PaperPosition',[0 0 1 1])

hold on;

set(gca,AXES\_DEFAULT\_PROPS{:})

axax = group\_violin(T\_vals\_plot,varnames{var\_i},'cond\_char','group\_char',...

fig,...

'VIOLIN\_PARAMS',VIOLIN\_PARAMS,...

'PLOT\_PARAMS',PLOT\_PARAMS,...

'STATS\_STRUCT',STATS\_STRUCT);

% set(axax,'OuterPosition',[0,0,1,1]);

% set(axax,'Position',[0.1+horiz\_shift,VIOLIN\_BOTTOM+vert\_shift,AX\_W\*im\_resize,AX\_H\*im\_resize]); %[left bottom width height]

hold off;

% exportgraphics(fig,[tmp\_savedir filesep sprintf('%s\_kinematics-speed-factor\_grouped.tiff',varnames{var\_i})],'Resolution',300)

% exportgraphics(fig,[tmp\_savedir filesep sprintf('%s\_kinematics-speed-contin\_grouped.tiff',varnames{var\_i})],'Resolution',300)

% close(fig)

%- iterate

end

end