Stats 506, F20, Problem Set 1

Zhihao Xu, xuzhihao@umich.edu

```
## ```SAS
```

```
## Warning in readLines("./ps6_q1.sas"): incomplete final line found on './
## ps6_q1.sas'
```

```
## /******************************************************************************/
## * Stats 506, Fall 2019                                                          ;
## * Problem Set 6, Question 1                                                     ;
## *------------------------------------------------------------------------------;
## * This script solves question 1 for Problem Set 6:                              ;
## * 1. Fit a linear mixed model to explore how each curvature measure differ      ;
## *     by condition                                                              ;
## *------------------------------------------------------------------------------;
## * Author: Jie Cao (caojie@umich.edu)                                        ;
## * Last updated on: Dec 10, 2019                                                 ;
## /******************************************************************************/
##
##
## * 80: --------------------------------------------------------------------------;
##
##
## /* Directories */
## libname ps6 "M:\506\hw\hw6";
##
## /* Import csv data */
## proc import datafile = "M:\506\hw\hw6\mousetrap_data.csv"
##  out = mouse_data
##  dbms = csv replace;
## run;
##
## /* Log transform four curvature measures */
## data mouse_data;
##  set mouse_data;
##  log_tot_dist = log(tot_dist);
##  log_max_abs_dev = log(max_abs_dev);
##  log_avg_abs_dev = log(avg_abs_dev);
##  log_AUC = log(AUC);
## run;
##
## /* Fit linear mixed model:
##     Response: log-tranformed curvature measure
##  Predictors: Condition
##  Random effect (intercept): subject, Exemplar*/
##
## /* Macro to loop over four measures and fit a LMM for each */
```

```
## %let y = tot_dist max_abs_dev avg_abs_dev AUC;
## %macro looplmm(vlist);
## %let i = 1;
## %do %while (%scan(&vlist., &i.) ne );
##  %let this_y = %scan(&vlist., &i.);
##
##  /* Drop observations with non-valid measure */
##  data model_data;
##      set mouse_data;
##      if log_&this_y. = . then delete;
##  run;
##
##  /* LMM for this curvature meature */
##  proc mixed data = model_data method = ML;
##      /* Factor variables */
##      class Condition(ref = "Typical") subject_nr Exemplar;
##      /* Model formular - response variable & fixed effect */
##      model log_&this_y. = Condition / solution;
##      /* Random intercepts */
##      random intercept / subject = subject_nr;
##      random intercept / subject = Exemplar;
##      /* Output estimates for fixed effect and covariance parameters */
##      ods output SolutionF = fe_&this_y. CovParms = re_&this_y.;
##  run;
##
##  /* Calculate relative effect and 95% CI for the fixed effect */
##  data fe_&this_y.(keep = Effect Condition re lci uci);
##      set fe_&this_y.;
##      where Condition = "Atypical";
##      re = exp(Estimate);
##      lci = exp(Estimate - 1.96 * StdErr);
##      uci = exp(Estimate + 1.96 * StdErr);
##  run;
##  /* Add measusre name to the dataset */
##  data fe_&this_y.;
##      set fe_&this_y.;
##      measure = "&this_y.";
##  run;
##
##  /* Calculate standard deviations for each variance component */
##  data re_&this_y.(drop = estimate);
##      set re_&this_y.;
##      if CovParm = "Residual" then Subject = "Error";
##      sd = sqrt(Estimate);
##  run;
## %let i = %eval(&i. + 1);
## %end;
## %mend;
## %looplmm(&y.);
##
## /* Put fixed effect results for each measure together */
## data fe_all (keep = measure relative_effect);
##  length measure $26.;
##  set fe_tot_dist(in = a)
##      fe_max_abs_dev(in = b)
##      fe_avg_abs_dev(in = c)
##      fe_auc(in = d);
##
##  if a then measure = "Total Distance";
##  if b then measure = "Maximum Absolute Deviation";
##  if c then measure = "Average Absolute Deviation";
##  if d then measure = "AUC";
##
##  relative_effect = cat(put(re, f4.2 -L), ' (',
##                        put(lci, f4.2 -L), ', ',
##                        put(uci, f4.2 -L), ')');
##
##  /* Add label to variables */
##  label measure = "Measure"
##        relative_effect = "Relative effect (95% CI)";
##
## run;
##
## /* Put standard deviation for each covariance component together */
## data re_all(drop = CovParm);
##  length measure $26.;
##  set re_tot_dist(in = a)
##      re_max_abs_dev(in = b)
##      re_avg_abs_dev(in = c)
##      re_auc(in = d);
##
##  if a then measure = "Total Distance";
##  if b then measure = "Maximum Absolute Deviation";
##  if c then measure = "Average Absolute Deviation";
##  if d then measure = "AUC";
## run;
##
## /* Transpose standard deviation table */
## proc sort data = re_all;
##  by measure;
## run;
## proc transpose data = re_all out = re_wide(drop = _name_);
##  by measure;
##  id Subject;
##  var sd;
## run;
## /* Add label to variable */
## data re_wide;
##  set re_wide;
##  label measure = "Measure"
##        subject_nr = "Subject"
##        Exemplar = "Exemplar"
##        Error = "Error";
## run;
##
## /* Merge two tables for a final output */
## proc sql;
##  create table q1_out as
##  select a.measure,
##         a.relative_effect,
##         b.subject_nr,
##         b.Exemplar,
##         b.Error
##  from fe_all a
##  left join re_wide b
##  on a.measure = b.measure;
## quit;
##
## /* Export output to a csv file */
## proc export data = q1_out dbms = csv
##  outfile = "M:\506\hw\hw6\ps6_q1.csv" label replace;
## run;

## ```
```