```
##Sink resykts to PDF file
pdf("OverallresutsFinal.pdf")

## load necessary libraries
library("reshape2")
library("plyr")
library("dplyr")
library("poLCA")
library("ggplot2")
library("ggparallel")
library("igraph")
library("tidyr")
library("knitr")
library("xtable")
library("forcats")
library("lattice")
library("Hmisc")
library("corrplot")
library("haven")
library("psych")
library("powerAnalysis")
library("RcmdrMisc")
library("logistf")
library("gmodels")
library("generalhoslem")
library("DescTools")

## Read in BFI data
FullSPSSData <-
  read_sav("Cultural Contribution SJY missing clusters and factors.sav")

## Set up variables and funtioncs to be used in the analyses

## Set up bic variables
max_II <- -100000
min_bic <- 0

##Set up function for entropy calculation
entropy <- function (var.p) {
  sum(-var.p * log(var.p))
}

##Function to find best result from 20 iterations
##to avoid local minima
##Return best LCA results and BIC values table
findbestlca.function <-
  function(analysis.vars,
           analysis.data,
           var.minclasses,
           var.maxclasses) {
    ##Set up temporary return, results and models frames
    bestnoclasses.var = 0
    lcaresults.list <- list()
    lcmodels.dataframe <- data.frame(
      NoClasses = 0,
      ll = 0,
      df = 0,
      BIC = 0,
      AIC = 0,
      ll_ratio = 0,
      Chi = 0,
      Entropy = 0
    )
```

```
##Loop to undertake LCA on min to max classes
for (i in seq(var.minclasses, var.maxclasses, 1)) {
  str.message <- paste("Running model for ", i, "classes")
  writeLines(str.message)
  lc <-
    poLCA(
      analysis.vars,
      analysis.data,
      nclass = i,
      maxiter = 5000,
      tol = 1e-5,
      nrep = 20,
      na.rm = TRUE,
      verbose = TRUE,
      calc.se = TRUE
    )
  ##Check for best model based on BIC criterion
  #If this is first model set min_bic to model BIC
  if (min_bic == 0) {
    min_bic <- lc$bic
  }
  ##If only one model sought then dont worry about min_bic, else check
  if (var.minclasses == var.maxclasses) {
    lca.best.model <- lc
    bestnoclasses.var = i
  } else if (lc$bic < min_bic) {
    min_bic <- lc$bic
    lca.best.model <- lc
    bestnoclasses.var = i
  }
  ##Store criteria for comparative table
  lcmodels.dataframe[i - 1, 1] <- i
  lcmodels.dataframe[i - 1, 2] <- lc$llik
  lcmodels.dataframe[i - 1, 3] <- lc$resid.df
  lcmodels.dataframe[i - 1, 4] <- lc$bic
  lcmodels.dataframe[i - 1, 5] <- lc$aic
  lcmodels.dataframe[i - 1, 6] <- lc$Gsq
  lcmodels.dataframe[i - 1, 7] <- lc$Chisq
  if (i > var.minclasses) {
    error_post <- mean(apply(lc$posterior, 1, entropy), na.rm = TRUE)
    lcmodels.dataframe[i - 1 , 8] <-
      round(((error_prior - error_post) / error_prior), 3)
    error_prior <- entropy(lc$P) # class proportions model 2
  } else {
    lcmodels.dataframe[i, 8] <- c("-")
    error_prior <- entropy(lc$P) # class proportions model 1
  }
}
##Force consistent order for classes
probs.start.new <-
  poLCA.reorder(lca.best.model$probs.start,
                order(lca.best.model$P, decreasing = TRUE))
lca.best.model <- poLCA(
  analysis.vars,
  analysis.data,
  nclass = bestnoclasses.var,
  maxiter = 5000,
  tol = 1e-5,
  na.rm = TRUE,
  verbose = FALSE,
  calc.se = TRUE,
  probs.start = probs.start.new
)
```

```
    ##Create return list
    lcaresults.list$models <- lcmodels.dataframe
    lcaresults.list$best <- lca.best.model
    lcaresults.list$bestnoclasses <- bestnoclasses.var
    writeLines("\nTable of BIC values for model selection\n")
    print(lcaresults.list$models)
    writeLines("\nBest Model From Analysis\n")
    print(lcaresults.list$best)
    return(lcaresults.list)
  }

##Function to plot variable probabilites by latent class
graphsfindbestlca.function <-
  function(lcmodel, str.title, str.fill, str.x) {
    #melt(lcaResults, level = 2)
    lcaresult.horizontal.plot <-
      ggplot(lcmodel, aes(x = L2, y = value, fill = X2)) +
      geom_bar(stat = "identity", position = "stack") +
      facet_grid(X1 ~ .) +
      scale_fill_brewer(type = "seq", palette = "Blues") +
      theme_bw() +
      ggtitle(str.title) +
      labs(fill = str.fill, x = str.x, y = "Share of item-\nresponse categories")
+
      theme(
        axis.text.y = element_blank(),
        axis.ticks.y = element_blank(),
        axis.text.x = element_text(
          angle = 90,
          vjust = 0.5,
          hjust = 1
        ),
        panel.grid.major.y = element_blank()
      ) +
      guides(fill = guide_legend(reverse = TRUE))
    print(lcaresult.horizontal.plot)
    lcaresult.vertical.plot <-
      ggplot(lcmodel, aes(x = X1, y = value, fill = X2)) +
      geom_bar(stat = "identity", position = "stack") +
      facet_wrap(~ L2) +
      scale_x_discrete(str.x, expand = c(0, 0)) +
      scale_y_continuous("Share of item-\nresponse categories", expand = c(0, 0))
+
      scale_fill_brewer(type = "seq", palette = "Blues") +
      theme_bw() +
      ggtitle(str.title) +
      labs(fill = str.fill) +
      theme(
        axis.text.y = element_blank(),
        axis.ticks.y = element_blank(),
        axis.text.x = element_text(
          angle = 90,
          vjust = 0.5,
          hjust = 1
        ),
        panel.grid.major.y = element_blank()
      ) +
      guides(fill = guide_legend(reverse = TRUE))
    print(lcaresult.vertical.plot)
  }

##Funciton to plot regresssion lines if regression requested
regressiongraph.function <-
  function(analysis.data,
```

```
            str.title,
            str.xaxistitle,
            var.range) {
    pidmat <- cbind(1, c(1:var.range))
    exb <- exp(pidmat %*% analysis.data)
    matplot(
      c(1:var.range),
      (cbind(1, exb) / (1 + rowSums(exb))),
      main = str.title,
      xlab = str.xaxistitle,
      ylab = "Probability of latent class membership",
      ylim = c(0, 1),
      type = "l",
      lwd = 3,
      col = 1:3
    )
    legend(
      "topleft",
      inset = .05,
      legend = c(str.c1, str.c2, str.c3),
      pch = 1,
      col = 1:3,
      horiz = TRUE
    )
  }

##Plot BIC and relted values
plotbicandentropy.function <- function(models) {
  results.melt <-
    melt(models, id = "NoClasses")
  results.melt
  models.plot <- ggplot(results.melt) +
    geom_point(aes(x = NoClasses, y = value), size = 2) +
    geom_line(aes(NoClasses, value, group = 1)) +
    theme_bw() +
    labs(x = "", y = "", title = "") +
    facet_grid(variable ~ ., scales = "free") +
    theme_bw(base_size = 8, base_family = "") +
    theme(
      panel.grid.major.x = element_blank() ,
      panel.grid.major.y = element_line(colour = "grey", size = 0.5),
      legend.title = element_text(size = 8, face = 'bold'),
      axis.text = element_text(size = 8),
      axis.title = element_text(size = 8),
      legend.text =  element_text(size = 8),
      axis.line = element_line(colour = "black"),
      strip.text.y = element_text(size = 8)
    )
  print(models.plot)
}

##Function to create regression results table if regressions requested
regressionresults.function <- function(lcamodel, str.regname) {
  var.numberclasses <- length(lcamodel$P)
  var.checkforregression <-
    ifelse(is.na(lcamodel$coeff[1]), 1, nrow(lcamodel$coeff))

  if (var.checkforregression > 1) {
    i = 0
    for (r in 2:var.numberclasses) {
      i = i + 1
      regression.table <- data.frame(
        ##reg =  paste(r, "/ 1"),
        coeff = round(lcamodel$coeff[, (r - 1)], 5),
```

```
        se = round(lcamodel$coeff.se[, (r - 1)], 5),
        tval = round(lcamodel$coeff[, (r - 1)] / lcamodel$coeff.se[, (r -
                                                                      1)], 3),
        pr = round(1 - (2 * abs(
          pt(lcamodel$coeff[, (r - 1)] / lcamodel$coeff.se[, (r - 1)],
lcamodel$resid.df) - 0.5
        )), 3)
      )
      str.filename <-
        paste(getwd(),
              "Regression",
              str.regname,
              r,
              "-1.latex",
              sep = "")
      print(xtable(regression.table), file = str.filename)
      str.filename <-
        paste(getwd(),
              "Regression",
              str.regname,
              r,
              "-1.htm",
              sep = "")
      print(xtable(regression.table),
            type = "html",
            file = str.filename)
    }

  }
}

## Function to undertake chisquare analayis and plot graphs of residuals and
contributions
chisquaretest.predictions.function <-
  function(indfactor.data,
           predclass.data,
           noclasses,
           pitem,
           gitem,
           chirows,
           chicols) {
    chisquare.results <- chisq.test(indfactor.data, predclass.data)
    residuals.data <- chisquare.results$residuals
    colnames(residuals.data) <- chicols
    rownames(residuals.data) <- chirows
    title.text <-
      paste(
        "Residuals: chi Square Crosstabulation of\n",
        pitem,
        "and",
        gitem,
        "\n(Chisquare =",
        round(chisquare.results$statistic, 3),
        " p <",
        round(chisquare.results$p.value, 3),
        ")",
        sep = " "
      )
    corrplot(
      residuals.data,
      is.cor = FALSE,
      title = title.text,
      mar = c(0, 0, 4, 0)
    )
```

```
    contrib.data <-
      100 * residuals.data ^ 2 / chisquare.results$statistic
    round(contrib.data, 3)
    colnames(contrib.data) <- chicols
    rownames(contrib.data) <- chirows
    title.text <-
      title.text <-
      paste(
        "Contributions: chi Square Crosstabulation of\n",
        pitem,
        "and",
        gitem,
        "\n(Chisquare =",
        round(chisquare.results$statistic, 3),
        " p <",
        round(chisquare.results$p.value, 3),
        ")",
        sep = " "
      )
    corrplot(
      contrib.data,
      is.cor = FALSE,
      title = title.text,
      mar = c(0, 0, 4, 0)
    )
    return(chisquare.results)
  }

##Funciton for Cramers V test
cv.test = function(x, y) {
  CV = sqrt(chisq.test(x, y, correct = FALSE)$statistic /
              (length(x) * (min(
                length(unique(x)), length(unique(y))
              ) - 1)))
  print.noquote("Cramér V / Phi:")
  return(as.numeric(CV))
}


##Lists of data items for use in recoding and analysis
lcaDataPerceptionsRecode <-
  FullSPSSData[, c(
    "qb5_1_1",
    "qb5_1_2",
    "qb5_1_3",
    "qb5_1_4",
    "qb5_1_5",
    "qb5_1_6",
    "qb5_1_7",
    "qb5_1_8",
    "qb5_1_9",
    "qb5_1_10",
    "qb5_1_11",
    "qb5_1_12",
    "qb5_1_13",
    "qb5_1_14",
    "qb5_1_15",
    "qb5_1_16",
    "qb5_1_17",
    "qb5_1_18",
    "qb5_2_1",
    "qb5_2_2",
    "qb5_2_3",
    "qb5_2_4",
```

```
"qb5_2_5",
"qb5_2_6",
"qb5_2_7",
"qb5_2_8",
"qb5_2_9",
"qb5_2_10",
"qb5_2_11",
"qb5_2_12",
"qb5_2_13",
"qb5_2_14",
"qb5_2_15",
"qb5_2_16",
"qb5_2_17",
"qb5_2_18",
"qb5_3_1",
"qb5_3_2",
"qb5_3_3",
"qb5_3_4",
"qb5_3_5",
"qb5_3_6",
"qb5_3_7",
"qb5_3_8",
"qb5_3_9",
"qb5_3_10",
"qb5_3_11",
"qb5_3_12",
"qb5_3_13",
"qb5_3_14",
"qb5_3_15",
"qb5_3_16",
"qb5_3_17",
"qb5_3_18",
"qb5_4_1",
"qb5_4_2",
"qb5_4_3",
"qb5_4_4",
"qb5_4_5",
"qb5_4_6",
"qb5_4_7",
"qb5_4_8",
"qb5_4_9",
"qb5_4_10",
"qb5_4_11",
"qb5_4_12",
"qb5_4_13",
"qb5_4_14",
"qb5_4_15",
"qb5_4_16",
"qb5_4_17",
"qb5_4_18",
"qb5_5_1",
"qb5_5_2",
"qb5_5_3",
"qb5_5_4",
"qb5_5_5",
"qb5_5_6",
"qb5_5_7",
"qb5_5_8",
"qb5_5_9",
"qb5_5_10",
"qb5_5_11",
"qb5_5_12",
"qb5_5_13",
"qb5_5_14",
```

```
"qb5_5_15",
"qb5_5_16",
"qb5_5_17",
"qb5_5_18",
"qb5_6_1",
"qb5_6_2",
"qb5_6_3",
"qb5_6_4",
"qb5_6_5",
"qb5_6_6",
"qb5_6_7",
"qb5_6_8",
"qb5_6_9",
"qb5_6_10",
"qb5_6_11",
"qb5_6_12",
"qb5_6_13",
"qb5_6_14",
"qb5_6_15",
"qb5_6_16",
"qb5_6_17",
"qb5_6_18",
"qb5_7_1",
"qb5_7_2",
"qb5_7_3",
"qb5_7_4",
"qb5_7_5",
"qb5_7_6",
"qb5_7_7",
"qb5_7_8",
"qb5_7_9",
"qb5_7_10",
"qb5_7_11",
"qb5_7_12",
"qb5_7_13",
"qb5_7_14",
"qb5_7_15",
"qb5_7_16",
"qb5_7_17",
"qb5_7_18",
"qb5_8_1",
"qb5_8_2",
"qb5_8_3",
"qb5_8_4",
"qb5_8_5",
"qb5_8_6",
"qb5_8_7",
"qb5_8_8",
"qb5_8_9",
"qb5_8_10",
"qb5_8_11",
"qb5_8_12",
"qb5_8_13",
"qb5_8_14",
"qb5_8_15",
"qb5_8_16",
"qb5_8_17",
"qb5_8_18",
"qb5_9_1",
"qb5_9_2",
"qb5_9_3",
"qb5_9_4",
"qb5_9_5",
"qb5_9_6",
```

```
"qb5_9_7",
"qb5_9_8",
"qb5_9_9",
"qb5_9_10",
"qb5_9_11",
"qb5_9_12",
"qb5_9_13",
"qb5_9_14",
"qb5_9_15",
"qb5_9_16",
"qb5_9_17",
"qb5_9_18",
"qb5_10_1",
"qb5_10_2",
"qb5_10_3",
"qb5_10_4",
"qb5_10_5",
"qb5_10_6",
"qb5_10_7",
"qb5_10_8",
"qb5_10_9",
"qb5_10_10",
"qb5_10_11",
"qb5_10_12",
"qb5_10_13",
"qb5_10_14",
"qb5_10_15",
"qb5_10_16",
"qb5_10_17",
"qb5_10_18",
"qb5_11_1",
"qb5_11_2",
"qb5_11_3",
"qb5_11_4",
"qb5_11_5",
"qb5_11_6",
"qb5_11_7",
"qb5_11_8",
"qb5_11_9",
"qb5_11_10",
"qb5_11_11",
"qb5_11_12",
"qb5_11_13",
"qb5_11_14",
"qb5_11_15",
"qb5_11_16",
"qb5_11_17",
"qb5_11_18",
"qb5_12_1",
"qb5_12_2",
"qb5_12_3",
"qb5_12_4",
"qb5_12_5",
"qb5_12_6",
"qb5_12_7",
"qb5_12_8",
"qb5_12_9",
"qb5_12_10",
"qb5_12_11",
"qb5_12_12",
"qb5_12_13",
"qb5_12_14",
"qb5_12_15",
"qb5_12_16",
```

```
"qb5_12_17",
"qb5_12_18",
"qb5_13_1",
"qb5_13_2",
"qb5_13_3",
"qb5_13_4",
"qb5_13_5",
"qb5_13_6",
"qb5_13_7",
"qb5_13_8",
"qb5_13_9",
"qb5_13_10",
"qb5_13_11",
"qb5_13_12",
"qb5_13_13",
"qb5_13_14",
"qb5_13_15",
"qb5_13_16",
"qb5_13_17",
"qb5_13_18",
"qb5_14_1",
"qb5_14_2",
"qb5_14_3",
"qb5_14_4",
"qb5_14_5",
"qb5_14_6",
"qb5_14_7",
"qb5_14_8",
"qb5_14_9",
"qb5_14_10",
"qb5_14_11",
"qb5_14_12",
"qb5_14_13",
"qb5_14_14",
"qb5_14_15",
"qb5_14_16",
"qb5_14_17",
"qb5_14_18",
"qb5_15_1",
"qb5_15_2",
"qb5_15_3",
"qb5_15_4",
"qb5_15_5",
"qb5_15_6",
"qb5_15_7",
"qb5_15_8",
"qb5_15_9",
"qb5_15_10",
"qb5_15_11",
"qb5_15_12",
"qb5_15_13",
"qb5_15_14",
"qb5_15_15",
"qb5_15_16",
"qb5_15_17",
"qb5_15_18",
"ClusterArtHouse",
"ClusterRomance",
"ClusterPopular",
"ClusterDocumentary",
"ClusterSFFantasy",
"ClusterMusicals",
"ClusterAnimation",
"ClusterFamily",
```

```
    "ClusterComic",
    "ClusterHorror"
  )] + 1

colnames(lcaDataPerceptionsRecode) <-
  c(
    "Film_Entertaining",
    "Film_Sociable",
    "Film_Educational",
    "Film_Relaxing",
    "Film_Escapism",
    "Film_Thought_provoking",
    "Film_Rewarding",
    "Film_Self_development",
    "Film_Fashionable",
    "Film_Inspirational",
    "Film_Emotional",
    "Film_Well_being",
    "Film_Boring",
    "Film_Negative",
    "Film_Artistic_value",
    "Film_Exciting",
    "Film_None",
    "Film_Dont_know",
    "Classical_music_Entertaining",
    "Classical_music_Sociable",
    "Classical_music_Educational",
    "Classical_music_Relaxing",
    "Classical_music_Escapism",
    "Classical_music_Thought_provoking",
    "Classical_music_Rewarding",
    "Classical_music_Self_development",
    "Classical_music_Fashionable",
    "Classical_music_Inspirational",
    "Classical_music_Emotional",
    "Classical_music_Well_being",
    "Classical_music_Boring",
    "Classical_music_Negative",
    "Classical_music_Artistic_value",
    "Classical_music_Exciting",
    "Classical_music_None",
    "Classical_music_Dont_know",
    "Pop_Entertaining",
    "Pop_Sociable",
    "Pop_Educational",
    "Pop_Relaxing",
    "Pop_Escapism",
    "Pop_Thought_provoking",
    "Pop_Rewarding",
    "Pop_Self_development",
    "Pop_Fashionable",
    "Pop_Inspirational",
    "Pop_Emotional",
    "Pop_Well_being",
    "Pop_Boring",
    "Pop_Negative",
    "Pop_Artistic_value",
    "Pop_Exciting",
    "Pop_None",
    "Pop_Dont_know",
    "Television_Entertaining",
    "Television_Sociable",
    "Television_Educational",
    "Television_Relaxing",
```

```
"Television_Escapism",
"Television_Thought_provoking",
"Television_Rewarding",
"Television_Self_development",
"Television_Fashionable",
"Television_Inspirational",
"Television_Emotional",
"Television_Well_being",
"Television_Boring",
"Television_Negative",
"Television_Artistic_value",
"Television_Exciting",
"Television_None",
"Television_Dont_know",
"Theatre_dance_Entertaining",
"Theatre_dance_Sociable",
"Theatre_dance_Educational",
"Theatre_dance_Relaxing",
"Theatre_dance_Escapism",
"Theatre_dance_Thought_provoking",
"Theatre_dance_Rewarding",
"Theatre_dance_Self_development",
"Theatre_dance_Fashionable",
"Theatre_dance_Inspirational",
"Theatre_dance_Emotional",
"Theatre_dance_Well_being",
"Theatre_dance_Boring",
"Theatre_dance_Negative",
"Theatre_dance_Artistic_value",
"Theatre_dance_Exciting",
"Theatre_dance_None",
"Theatre_dance_Dont_know",
"Literature_Entertaining",
"Literature_Sociable",
"Literature_Educational",
"Literature_Relaxing",
"Literature_Escapism",
"Literature_Thought_provoking",
"Literature_Rewarding",
"Literature_Self_development",
"Literature_Fashionable",
"Literature_Inspirational",
"Literature_Emotional",
"Literature_Well_being",
"Literature_Boring",
"Literature_Negative",
"Literature_Artistic_value",
"Literature_Exciting",
"Literature_None",
"Literature_Dont_know",
"News_newspapers_Entertaining",
"News_newspapers_Sociable",
"News_newspapers_Educational",
"News_newspapers_Relaxing",
"News_newspapers_Escapism",
"News_newspapers_Thought_provoking",
"News_newspapers_Rewarding",
"News_newspapers_Self_development",
"News_newspapers_Fashionable",
"News_newspapers_Inspirational",
"News_newspapers_Emotional",
"News_newspapers_Well_being",
"News_newspapers_Boring",
"News_newspapers_Negative",
```

```
"News_newspapers_Artistic_value",
"News_newspapers_Exciting",
"News_newspapers_None",
"News_newspapers_Dont_know",
"Art_galleries_museums_Entertaining",
"Art_galleries_museums_Sociable",
"Art_galleries_museums_Educational",
"Art_galleries_museums_Relaxing",
"Art_galleries_museums_Escapism",
"Art_galleries_museums_Thought_provoking",
"Art_galleries_museums_Rewarding",
"Art_galleries_museums_Self_development",
"Art_galleries_museums_Fashionable",
"Art_galleries_museums_Inspirational",
"Art_galleries_museums_Emotional",
"Art_galleries_museums_Well_being",
"Art_galleries_museums_Boring",
"Art_galleries_museums_Negative",
"Art_galleries_museums_Artistic_value",
"Art_galleries_museums_Exciting",
"Art_galleries_museums_None",
"Art_galleries_museums_Dont_know",
"Videogames_Entertaining",
"Videogames_Sociable",
"Videogames_Educational",
"Videogames_Relaxing",
"Videogames_Escapism",
"Videogames_Thought_provoking",
"Videogames_Rewarding",
"Videogames_Self_development",
"Videogames_Fashionable",
"Videogames_Inspirational",
"Videogames_Emotional",
"Videogames_Well_being",
"Videogames_Boring",
"Videogames_Negative",
"Videogames_Artistic_value",
"Videogames_Exciting",
"Videogames_None",
"Videogames_Dont_know",
"Watching_sport_Entertaining",
"Watching_sport_Sociable",
"Watching_sport_Educational",
"Watching_sport_Relaxing",
"Watching_sport_Escapism",
"Watching_sport_Thought_provoking",
"Watching_sport_Rewarding",
"Watching_sport_Self_development",
"Watching_sport_Fashionable",
"Watching_sport_Inspirational",
"Watching_sport_Emotional",
"Watching_sport_Well_being",
"Watching_sport_Boring",
"Watching_sport_Negative",
"Watching_sport_Artistic_value",
"Watching_sport_Exciting",
"Watching_sport_None",
"Watching_sport_Dont_know",
"Doing_sport_exercise_Entertaining",
"Doing_sport_exercise_Sociable",
"Doing_sport_exercise_Educational",
"Doing_sport_exercise_Relaxing",
"Doing_sport_exercise_Escapism",
"Doing_sport_exercise_Thought_provoking",
```

```
"Doing_sport_exercise_Rewarding",
"Doing_sport_exercise_Self_development",
"Doing_sport_exercise_Fashionable",
"Doing_sport_exercise_Inspirational",
"Doing_sport_exercise_Emotional",
"Doing_sport_exercise_Well_being",
"Doing_sport_exercise_Boring",
"Doing_sport_exercise_Negative",
"Doing_sport_exercise_Artistic_value",
"Doing_sport_exercise_Exciting",
"Doing_sport_exercise_None",
"Doing_sport_exercise_Dont_know",
"Pubs_clubs_Entertaining",
"Pubs_clubs_Sociable",
"Pubs_clubs_Educational",
"Pubs_clubs_Relaxing",
"Pubs_clubs_Escapism",
"Pubs_clubs_Thought_provoking",
"Pubs_clubs_Rewarding",
"Pubs_clubs_Self_development",
"Pubs_clubs_Fashionable",
"Pubs_clubs_Inspirational",
"Pubs_clubs_Emotional",
"Pubs_clubs_Well_being",
"Pubs_clubs_Boring",
"Pubs_clubs_Negative",
"Pubs_clubs_Artistic_value",
"Pubs_clubs_Exciting",
"Pubs_clubs_None",
"Pubs_clubs_Dont_know",
"Restaurants_Entertaining",
"Restaurants_Sociable",
"Restaurants_Educational",
"Restaurants_Relaxing",
"Restaurants_Escapism",
"Restaurants_Thought_provoking",
"Restaurants_Rewarding",
"Restaurants_Self_development",
"Restaurants_Fashionable",
"Restaurants_Inspirational",
"Restaurants_Emotional",
"Restaurants_Well_being",
"Restaurants_Boring",
"Restaurants_Negative",
"Restaurants_Artistic_value",
"Restaurants_Exciting",
"Restaurants_None",
"Restaurants_Dont_know",
"Countryside_Entertaining",
"Countryside_Sociable",
"Countryside_Educational",
"Countryside_Relaxing",
"Countryside_Escapism",
"Countryside_Thought_provoking",
"Countryside_Rewarding",
"Countryside_Self_development",
"Countryside_Fashionable",
"Countryside_Inspirational",
"Countryside_Emotional",
"Countryside_Well_being",
"Countryside_Boring",
"Countryside_Negative",
"Countryside_Artistic_value",
"Countryside_Exciting",
```

```
    "Countryside_None",
    "Countryside_Dont_know",
    "Religion_Entertaining",
    "Religion_Sociable",
    "Religion_Educational",
    "Religion_Relaxing",
    "Religion_Escapism",
    "Religion_Thought_provoking",
    "Religion_Rewarding",
    "Religion_Self_development",
    "Religion_Fashionable",
    "Religion_Inspirational",
    "Religion_Emotional",
    "Religion_Well_being",
    "Religion_Boring",
    "Religion_Negative",
    "Religion_Artistic_value",
    "Religion_Exciting",
    "Religion_None",
    "Religion_Dont_know",
    "Art_house_and_foreign",
    "Romance",
    "Popular",
    "Documentary",
    "SF_Fantasy",
    "Musicals",
    "Animation",
    "Family",
    "Comic",
    "Horror"
  )

##Setup variables for LCA analyses
lca.perceptions.variables.list <- list()
lca.perceptions.variables.list$Film <- cbind(
  Film_Entertaining,
  Film_Sociable,
  Film_Educational,
  Film_Relaxing,
  Film_Escapism,
  Film_Thought_provoking,
  Film_Rewarding,
  Film_Self_development,
  Film_Fashionable,
  Film_Inspirational,
  Film_Emotional,
  Film_Well_being,
  Film_Boring,
  Film_Negative,
  Film_Artistic_value,
  Film_Exciting,
  Film_None,
  Film_Dont_know
) ~ 1

lca.perceptions.variables.list$Classical <- cbind(
  Classical_music_Entertaining,
  Classical_music_Sociable,
  Classical_music_Educational,
  Classical_music_Relaxing,
  Classical_music_Escapism,
  Classical_music_Thought_provoking,
  Classical_music_Rewarding,
  Classical_music_Self_development,
```

```
  Classical_music_Fashionable,
  Classical_music_Inspirational,
  Classical_music_Emotional,
  Classical_music_Well_being,
  Classical_music_Boring,
  Classical_music_Negative,
  Classical_music_Artistic_value,
  Classical_music_Exciting,
  Classical_music_None,
  Classical_music_Dont_know
) ~ 1

lca.perceptions.variables.list$Pop <- cbind(
  Pop_Entertaining,
  Pop_Sociable,
  Pop_Educational,
  Pop_Relaxing,
  Pop_Escapism,
  Pop_Thought_provoking,
  Pop_Rewarding,
  Pop_Self_development,
  Pop_Fashionable,
  Pop_Inspirational,
  Pop_Emotional,
  Pop_Well_being,
  Pop_Boring,
  Pop_Negative,
  Pop_Artistic_value,
  Pop_Exciting,
  Pop_None,
  Pop_Dont_know
) ~ 1

lca.perceptions.variables.list$TV <- cbind(
  Television_Entertaining,
  Television_Sociable,
  Television_Educational,
  Television_Relaxing,
  Television_Escapism,
  Television_Thought_provoking,
  Television_Rewarding,
  Television_Self_development,
  Television_Fashionable,
  Television_Inspirational,
  Television_Emotional,
  Television_Well_being,
  Television_Boring,
  Television_Negative,
  Television_Artistic_value,
  Television_Exciting,
  Television_None,
  Television_Dont_know
) ~ 1

lca.perceptions.variables.list$Theatre <- cbind(
  Theatre_dance_Entertaining,
  Theatre_dance_Sociable,
  Theatre_dance_Educational,
  Theatre_dance_Relaxing,
  Theatre_dance_Escapism,
  Theatre_dance_Thought_provoking,
  Theatre_dance_Rewarding,
  Theatre_dance_Self_development,
  Theatre_dance_Fashionable,
```

```
  Theatre_dance_Inspirational,
  Theatre_dance_Emotional,
  Theatre_dance_Well_being,
  Theatre_dance_Boring,
  Theatre_dance_Negative,
  Theatre_dance_Artistic_value,
  Theatre_dance_Exciting,
  Theatre_dance_None,
  Theatre_dance_Dont_know
) ~ 1

lca.perceptions.variables.list$Literature <- cbind(
  Literature_Entertaining,
  Literature_Sociable,
  Literature_Educational,
  Literature_Relaxing,
  Literature_Escapism,
  Literature_Thought_provoking,
  Literature_Rewarding,
  Literature_Self_development,
  Literature_Fashionable,
  Literature_Inspirational,
  Literature_Emotional,
  Literature_Well_being,
  Literature_Boring,
  Literature_Negative,
  Literature_Artistic_value,
  Literature_Exciting,
  Literature_None,
  Literature_Dont_know
) ~ 1

lca.perceptions.variables.list$News_newspapers <- cbind(
  News_newspapers_Entertaining,
  News_newspapers_Sociable,
  News_newspapers_Educational,
  News_newspapers_Relaxing,
  News_newspapers_Escapism,
  News_newspapers_Thought_provoking,
  News_newspapers_Rewarding,
  News_newspapers_Self_development,
  News_newspapers_Fashionable,
  News_newspapers_Inspirational,
  News_newspapers_Emotional,
  News_newspapers_Well_being,
  News_newspapers_Boring,
  News_newspapers_Negative,
  News_newspapers_Artistic_value,
  News_newspapers_Exciting,
  News_newspapers_None,
  News_newspapers_Dont_know
) ~ 1

lca.perceptions.variables.list$Art_galleries_museums <- cbind(
  Art_galleries_museums_Entertaining,
  Art_galleries_museums_Sociable,
  Art_galleries_museums_Educational,
  Art_galleries_museums_Relaxing,
  Art_galleries_museums_Escapism,
  Art_galleries_museums_Thought_provoking,
  Art_galleries_museums_Rewarding,
  Art_galleries_museums_Self_development,
  Art_galleries_museums_Fashionable,
  Art_galleries_museums_Inspirational,
```

```
  Art_galleries_museums_Emotional,
  Art_galleries_museums_Well_being,
  Art_galleries_museums_Boring,
  Art_galleries_museums_Negative,
  Art_galleries_museums_Artistic_value,
  Art_galleries_museums_Exciting,
  Art_galleries_museums_None,
  Art_galleries_museums_Dont_know
) ~ 1

lca.perceptions.variables.list$Videogames <- cbind(
  Videogames_Entertaining,
  Videogames_Sociable,
  Videogames_Educational,
  Videogames_Relaxing,
  Videogames_Escapism,
  Videogames_Thought_provoking,
  Videogames_Rewarding,
  Videogames_Self_development,
  Videogames_Fashionable,
  Videogames_Inspirational,
  Videogames_Emotional,
  Videogames_Well_being,
  Videogames_Boring,
  Videogames_Negative,
  Videogames_Artistic_value,
  Videogames_Exciting,
  Videogames_None,
  Videogames_Dont_know
) ~ 1


lca.perceptions.variables.list$Watching_sport <- cbind(
  Watching_sport_Entertaining,
  Watching_sport_Sociable,
  Watching_sport_Educational,
  Watching_sport_Relaxing,
  Watching_sport_Escapism,
  Watching_sport_Thought_provoking,
  Watching_sport_Rewarding,
  Watching_sport_Self_development,
  Watching_sport_Fashionable,
  Watching_sport_Inspirational,
  Watching_sport_Emotional,
  Watching_sport_Well_being,
  Watching_sport_Boring,
  Watching_sport_Negative,
  Watching_sport_Artistic_value,
  Watching_sport_Exciting,
  Watching_sport_None,
  Watching_sport_Dont_know
) ~ 1

lca.perceptions.variables.list$Doing_sport_exercise <- cbind(
  Doing_sport_exercise_Entertaining,
  Doing_sport_exercise_Sociable,
  Doing_sport_exercise_Educational,
  Doing_sport_exercise_Relaxing,
  Doing_sport_exercise_Escapism,
  Doing_sport_exercise_Thought_provoking,
  Doing_sport_exercise_Rewarding,
  Doing_sport_exercise_Self_development,
  Doing_sport_exercise_Fashionable,
  Doing_sport_exercise_Inspirational,
```

```
  Doing_sport_exercise_Emotional,
  Doing_sport_exercise_Well_being,
  Doing_sport_exercise_Boring,
  Doing_sport_exercise_Negative,
  Doing_sport_exercise_Artistic_value,
  Doing_sport_exercise_Exciting,
  Doing_sport_exercise_None,
  Doing_sport_exercise_Dont_know
) ~ 1

lca.perceptions.variables.list$Pubs_clubs <- cbind(
  Pubs_clubs_Entertaining,
  Pubs_clubs_Sociable,
  Pubs_clubs_Educational,
  Pubs_clubs_Relaxing,
  Pubs_clubs_Escapism,
  Pubs_clubs_Thought_provoking,
  Pubs_clubs_Rewarding,
  Pubs_clubs_Self_development,
  Pubs_clubs_Fashionable,
  Pubs_clubs_Inspirational,
  Pubs_clubs_Emotional,
  Pubs_clubs_Well_being,
  Pubs_clubs_Boring,
  Pubs_clubs_Negative,
  Pubs_clubs_Artistic_value,
  Pubs_clubs_Exciting,
  Pubs_clubs_None,
  Pubs_clubs_Dont_know
) ~ 1

lca.perceptions.variables.list$Restaurants <- cbind(
  Restaurants_Entertaining,
  Restaurants_Sociable,
  Restaurants_Educational,
  Restaurants_Relaxing,
  Restaurants_Escapism,
  Restaurants_Thought_provoking,
  Restaurants_Rewarding,
  Restaurants_Self_development,
  Restaurants_Fashionable,
  Restaurants_Inspirational,
  Restaurants_Emotional,
  Restaurants_Well_being,
  Restaurants_Boring,
  Restaurants_Negative,
  Restaurants_Artistic_value,
  Restaurants_Exciting,
  Restaurants_None,
  Restaurants_Dont_know
) ~ 1

lca.perceptions.variables.list$Countryside <- cbind(
  Countryside_Entertaining,
  Countryside_Sociable,
  Countryside_Educational,
  Countryside_Relaxing,
  Countryside_Escapism,
  Countryside_Thought_provoking,
  Countryside_Rewarding,
  Countryside_Self_development,
  Countryside_Fashionable,
  Countryside_Inspirational,
  Countryside_Emotional,
```

```
  Countryside_Well_being,
  Countryside_Boring,
  Countryside_Negative,
  Countryside_Artistic_value,
  Countryside_Exciting,
  Countryside_None,
  Countryside_Dont_know
) ~ 1

lca.perceptions.variables.list$Religion <- cbind(
  Religion_Entertaining,
  Religion_Sociable,
  Religion_Educational,
  Religion_Relaxing,
  Religion_Escapism,
  Religion_Thought_provoking,
  Religion_Rewarding,
  Religion_Self_development,
  Religion_Fashionable,
  Religion_Inspirational,
  Religion_Emotional,
  Religion_Well_being,
  Religion_Boring,
  Religion_Negative,
  Religion_Artistic_value,
  Religion_Exciting,
  Religion_None,
  Religion_Dont_know
) ~ 1

perceptions.group.list = c(
  "Film",
  "Classical",
  "Pop",
  "TV",
  "Theatre",
  "Literature",
  "News_newspapers",
  "Art_galleries_museums",
  "Videogames",
  "Watching_sport",
  "Doing_sport_exercise",
  "Pubs_clubs",
  "Restaurants",
  "Countryside",
  "Religion"
)

genre.group.list <- c(
  "Art_house_and_foreign",
  "Romance",
  "Popular",
  "Documentary",
  "SF_Fantasy",
  "Musicals",
  "Animation",
  "Family",
  "Comic",
  "Horror"
)

##Set up varibales for films watched
lcaDataFilmswatchedRecode <- FullSPSSData[, c(
  "qc3a_1",
```

```
  "qc3a_2",
  "qc3a_3",
  "qc3a_4",
  "qc3a_5",
  "qc3a_6",
  "qc3a_7",
  "qc3b_1",
  "qc3b_2",
  "qc3b_3",
  "qc3b_4",
  "qc3b_5",
  "qc3b_6",
  "qc3b_7",
  "qc3c_1",
  "qc3c_2",
  "qc3c_3",
  "qc3c_4",
  "qc3c_5",
  "qc3c_6",
  "qc3c_7",
  "qc3d_1",
  "qc3d_2",
  "qc3d_3",
  "qc3d_4",
  "qc3d_5",
  "qc3d_6",
  "qc3d_7",
  "qc3e_1",
  "qc3e_2",
  "qc3e_3",
  "qc3e_4",
  "qc3e_5",
  "qc3e_6",
  "qc3e_7",
  "qc3f_1",
  "qc3f_2",
  "qc3f_3",
  "qc3f_4",
  "qc3f_5",
  "qc3f_6",
  "qc3f_7"
  # "ViewedBlockbuster",
  # "ViewedFamouscast",
  # "ViewedIndependent",
  # "ViewedForeignlanguage",
  # "ViewedAnimated",
  # "ViewedOther",
  # "ViewedDontknow"
)] + 1

colnames(lcaDataFilmswatchedRecode) <- c(
  "Cinema_Blockbuster",
  "Cinema_Famous_cast",
  "Cinema_Independent",
  "Cinema_Foreign_language",
  "Cinema_Animated",
  "Cinema_Other",
  "Cinema_Dont_know",
  "Television_Blockbuster",
  "Television_Famous_cast",
  "Television_Independent",
  "Television_Foreign_language",
  "Television_Animated",
  "Television_Other",
```

```
  "Television_Dont_know",
  "DVD_Blu_ray_Blockbuster",
  "DVD_Blu_ray_Famous_cast",
  "DVD_Blu_ray_Independent",
  "DVD_Blu_ray_Foreign_language",
  "DVD_Blu_ray_Animated",
  "DVD_Blu_ray_Other",
  "DVD_Blu_ray_Dont_know",
  "Downloading_streaming_on_theinternet_Blockbuster",
  "Downloading_streaming_on_theinternet_Famous_cast",
  "Downloading_streaming_on_theinternet_Independent",
  "Downloading_streaming_on_theinternet_Foreign_language",
  "Downloading_streaming_on_theinternet_Animated",
  "Downloading_streaming_on_theinternet_Other",
  "Downloading_streaming_on_theinternet_Dont_know",
  "On_mobildevice_Blockbuster",
  "On_mobildevice_Famous_cast",
  "On_mobildevice_Independent",
  "On_mobildevice_Foreign_language",
  "On_mobildevice_Animated",
  "On_mobildevice_Other",
  "On_mobildevice_Dont_know",
  "Shown_on_plane_Blockbuster",
  "Shown_on_plane_Famous_cast",
  "Shown_on_plane_Independent",
  "Shown_on_plane_Foreign_language",
  "Shown_on_plane_Animated",
  "Shown_on_plane_Other",
  "Shown_on_plane_Dont_know"
)

##Calcualte range of viewed films
FullSPSSData$Viewed_Blockbuster <-
  rowSums(FullSPSSData[, c("qc3a_1", "qc3b_1", "qc3c_1", "qc3d_1", "qc3e_1",
"qc3f_1")], na.rm = TRUE)
FullSPSSData$Viewed_Famouscast <-
  rowSums(FullSPSSData[, c("qc3a_2", "qc3b_2", "qc3c_2", "qc3d_2", "qc3e_2",
"qc3f_2")], na.rm = TRUE)
FullSPSSData$Viewed_Independent <-
  rowSums(FullSPSSData[, c("qc3a_3", "qc3b_3", "qc3c_3", "qc3d_3", "qc3e_3",
"qc3f_3")], na.rm = TRUE)
FullSPSSData$Viewed_Foreignlanguage <-
  rowSums(FullSPSSData[, c("qc3a_4", "qc3b_4", "qc3c_4", "qc3d_4", "qc3e_4",
"qc3f_4")], na.rm = TRUE)
FullSPSSData$Viewed_Animated <-
  rowSums(FullSPSSData[, c("qc3a_5", "qc3b_5", "qc3c_5", "qc3d_5", "qc3e_5",
"qc3f_5")], na.rm = TRUE)
FullSPSSData$Viewed_Other <-
  rowSums(FullSPSSData[, c("qc3a_6", "qc3b_6", "qc3c_6", "qc3d_6", "qc3e_6",
"qc3f_6")], na.rm = TRUE)
FullSPSSData$Viewed_Dontknow <-
  rowSums(FullSPSSData[, c("qc3a_7", "qc3b_7", "qc3c_7", "qc3d_7", "qc3e_7",
"qc3f_7")], na.rm = TRUE)

##Create binary measure of films viewed
FullSPSSData$Viewed_Blockbuster[FullSPSSData$Viewed_Blockbuster > 0] <-
  1
FullSPSSData$Viewed_Famouscast[FullSPSSData$Viewed_Famouscast > 0] <-
  1

FullSPSSData$Viewed_Independent[FullSPSSData$Viewed_Independent > 0] <-
  1
FullSPSSData$Viewed_Foreignlanguage[FullSPSSData$Viewed_Foreignlanguage >
                                      0] <- 1
```

```
FullSPSSData$Viewed_Animated[FullSPSSData$Viewed_Animated > 0] <- 1
FullSPSSData$Viewed_Other[FullSPSSData$Viewed_Other > 0] <- 1
FullSPSSData$Viewed_Dontknow[FullSPSSData$Viewed_Dontknow > 0] <- 1

##Set up LCA analysis variables
lcaDataFilmswatchedRecode$Viewed_Blockbuster <-
  FullSPSSData$Viewed_Blockbuster + 1
lcaDataFilmswatchedRecode$Viewed_Famouscast <-
  FullSPSSData$Viewed_Famouscast + 1
lcaDataFilmswatchedRecode$Viewed_Independent <-
  FullSPSSData$Viewed_Independent + 1
lcaDataFilmswatchedRecode$Viewed_Foreignlanguage <-
  FullSPSSData$Viewed_Foreignlanguage + 1
lcaDataFilmswatchedRecode$Viewed_Animated <-
  FullSPSSData$Viewed_Animated + 1
lcaDataFilmswatchedRecode$Viewed_Other <-
  FullSPSSData$Viewed_Other + 1
lcaDataFilmswatchedRecode$Viewed_Dontknow <-
  FullSPSSData$Viewed_Dontknow + 1

lca.filmswatched.variables <- cbind(
  Cinema_Blockbuster,
  Cinema_Famous_cast,
  Cinema_Independent,
  Cinema_Foreign_language ,
  Cinema_Animated,
  Cinema_Other,
  Television_Blockbuster,
  Television_Famous_cast,
  Television_Independent,
  Television_Foreign_language,
  Television_Animated,
  Television_Other,
  DVD_Blu_ray_Blockbuster,
  DVD_Blu_ray_Famous_cast,
  DVD_Blu_ray_Independent,
  DVD_Blu_ray_Foreign_language,
  DVD_Blu_ray_Animated,
  DVD_Blu_ray_Other,
  Downloading_streaming_on_theinternet_Blockbuster,
  Downloading_streaming_on_theinternet_Famous_cast,
  Downloading_streaming_on_theinternet_Independent,
  Downloading_streaming_on_theinternet_Foreign_language,
  Downloading_streaming_on_theinternet_Animated,
  Downloading_streaming_on_theinternet_Other
  # On_mobildevice_Blockbuster,
  # On_mobildevice_Famous_cast,
  # On_mobildevice_Independent,
  # On_mobildevice_Foreign_language,
  # On_mobildevice_Animated,
  # Shown_on_plane_Blockbuster,
  # Shown_on_plane_Famous_cast,
  # Shown_on_plane_Independent,
  # Shown_on_plane_Foreign_language,
  # Shown_on_plane_Animated
) ~ 1

##Set up liked genres varibales
lcaDataGenresLikedRecode <- FullSPSSData[, c(
  "qc1_1",
  "qc1_2",
  "qc1_3",
  "qc1_4",
  "qc1_5",
```

```
  "qc1_6",
  "qc1_7",
  "qc1_8",
  "qc1_9",
  "qc1_10",
  "qc1_11",
  "qc1_12",
  "qc1_13",
  "qc1_14",
  "qc1_15",
  "qc1_16",
  "qc1_17",
  "qc1_18",
  "qc1_19",
  "qc1_20",
  "qc1_21",
  "qc1_22",
  "qc1_23",
  "qc1_24"
)] + 1

colnames(lcaDataGenresLikedRecode) <- c(
  "Action_adventure",
  "Animation",
  "Art_house_films",
  "Comedy",
  "Comic_book",
  "Classic_films",
  "Documentary",
  "Drama",
  "Family_film",
  "Fantasy",
  "Foreign_language",
  "Horror",
  "Musicals",
  "Romance",
  "Romantic_comedy",
  "Sci_fi",
  "Suspense_thriller",
  "Westerns_cowboy",
  "Historical",
  "War",
  "Gangster",
  "Other",
  "None",
  "Dont_know"
)

lcaDataGenresLikedRecode[is.na(lcaDataGenresLikedRecode)] <- 1

lca.filmgenres.variables <- cbind(
  Action_adventure,
  Animation,
  Art_house_films,
  Comedy,
  Comic_book,
  Classic_films,
  Documentary,
  Drama,
  Family_film,
  Fantasy,
  Foreign_language,
  Horror,
  Musicals,
```

```
  Romance,
  Romantic_comedy,
  Sci_fi,
  Suspense_thriller,
  #Westerns_cowboy,
  #Historical,
  #War,
  #Gangster,
  #Other,
  None
  #Dont_know
) ~ 1

lca.filmtypeswatched.variables <- cbind(
  Viewed_Blockbuster,
  Viewed_Famouscast,
  Viewed_Independent,
  Viewed_Foreignlanguage,
  Viewed_Animated,
  Viewed_Other,
  Viewed_Dontknow
) ~ 1

filmtypeswatched.group.list <- c(
  "Viewed_Blockbuster",
  "Viewed_Famouscast",
  "Viewed_Independent",
  "Viewed_Foreignlanguage",
  "Viewed_Animated",
  "Viewed_Other",
  "Viewed_Dontknow"
)


##Generate results for basic classes for each set of perceptions
# LCAPerceptionFullResults <- list()
# for (pitem in perceptions.group.list) {
#   ##Call analysis with min 2 and max 10 classes
#   LCAPerceptionFullResults[[pitem]] <-
#     findbestlca.function(lca.perceptions.variables.list[[pitem]],
#                          lcaDataPerceptionsRecode,
#                          2,
#                          10)
#   ##Lattice plot of variable proportions by class
#   title <-
#     paste("Perception of ", pitem, "\nNo Regression", sep = "")
#   str.fill <- paste("Agree (2) - Disagree (1)")
#   str.x <- paste("Perceptions")
#   graphsfindbestlca.function(melt(LCAPerceptionFullResults[[pitem]]$best$probs,
level = 2),
#                              title,
#                              str.fill,
#                              str.x)
#
#   ##Add predicted class to SPSS file
#   FullSPSSData[[pitem]] <-
#     LCAPerceptionFullResults[[pitem]]$best$predclass
#
#   for (gitem in genre.group.list) {
#     LCAPerceptionFullResults[[pitem]][[gitem]]$Chisq <-
#       lcaDataPerceptionsRecode[[gitem]][lcaDataPerceptionsRecode[[gitem]] ==
#                                           -98.99] <- NA
#
#     predclass.data <- lcaDataPerceptionsRecode[[gitem]]
```

```
#     indfactor.data <- LCAPerceptionFullResults[[pitem]]$best$predclass
#     chirows <- c("Not favourite", "Favourite")
#     chisquaretest.predictions.function(
#       indfactor.data,
#       predclass.data,
#       LCAPerceptionFullResults[[pitem]]$bestnoclasses,
#       pitem,
#       gitem,
#       chirows
#     )
#   }
# }
# for (pitem in perceptions.group.list) {
#   ##Plot model selection variables
#   plotbicandentropy.function(LCAPerceptionFullResults[[pitem]]$models)
# }

##LCA of types of film watched by medium
LCAFilmTypesWatchedAcrossAllMediaResults <-
  findbestlca.function(lca.filmswatched.variables,
                       lcaDataFilmswatchedRecode,
                       2,
                       10)

FilmAllMediaBestProbs <-
  spread(melt(LCAFilmTypesWatchedAcrossAllMediaResults$best$probs, level = 2),
         "L2",
         "value")
savefile <- file.path(getwd(), "filmsallmediaLCA.csv")
write.csv(FilmAllMediaBestProbs, savefile)

##Lattice plot of variable proportions by class
title <-
  paste("Platforms for film consumption", "\nNo Regression", sep = "")
str.fill1 <- paste("Viewed (2) - Not Viewed (1)")
str.x1 <- paste("Film types by platform")
lcmodel2 <-
  melt(LCAFilmTypesWatchedAcrossAllMediaResults$best$probs, level = 2)
lcmodel2$X2 <-
  revalue(lcmodel2$X2,
          c(" Pr(2)" = "Probability viewed", " Pr(1)" = "Probability not viewed"))
graphsfindbestlca.function(lcmodel2,
                           title, str.fill1, str.x1)

##Plot model selection variables
plotbicandentropy.function(LCAFilmTypesWatchedAcrossAllMediaResults$models)

# ##Add predicted class to SPSS file
# FullSPSSData$Filmbymediaclass <-
#   LCAFilmTypesWatchedAcrossAllMediaResults$best$predclass

##LCA of types of film watched overall
LCAFilmTypesWatchedFullResults <-
  findbestlca.function(lca.filmtypeswatched.variables,
                       lcaDataFilmswatchedRecode,
                       2,
                       10)

FilmBestProbs <-
  spread(melt(LCAFilmTypesWatchedFullResults$best$probs, level = 2),
         "L2",
         "value")
savefile <- file.path(getwd(), "filmsLCA.csv")
write.csv(FilmBestProbs, savefile)
```

```
##Lattice plot of variable proportions by class
title <-
  paste("Types of film viewed", "\nNo Regression", sep = "")
str.fill <- paste("Viewed (2) - Not Viewed (1)")
str.x <- paste("Film types viewed")
lcmodel3 <-
  melt(LCAFilmTypesWatchedFullResults$best$probs, level = 2)
lcmodel3$X2 <-
  revalue(lcmodel3$X2,
          c("Pr(2)" = "Probability viewed", "Pr(1)" = "Probability not viewed"))
graphsfindbestlca.function(lcmodel3,
                           title, str.fill, str.x)

##Plot model selection variables
plotbicandentropy.function(LCAFilmTypesWatchedFullResults$models)

##Add predicted class to SPSS file
FullSPSSData$Filmwatchedclass <-
  LCAFilmTypesWatchedFullResults$best$predclass

##LCA of genres liked by medium
LCAFilmGenresLiked <-
  findbestlca.function(lca.filmgenres.variables,
                       lcaDataGenresLikedRecode,
                       2,
                       10)

FilmGenreProbs <-
  spread(melt(LCAFilmGenresLiked$best$probs, level = 2), "L2", "value")
savefile <- file.path(getwd(), "filmgenresLCA.csv")
write.csv(FilmGenreProbs, savefile)

##Lattice plot of variable proportions by class
title <-
  paste("Genres of film consumption", "\nNo Regression", sep = "")
str.fill1 <- paste("Liked (2) - Not Liked (1)")
str.x1 <- paste("Film genres liked")
lcmodel4 <- melt(LCAFilmGenresLiked$best$probs, level = 2)
lcmodel4$X2 <-
  revalue(lcmodel4$X2,
          c("Pr(2)" = "Probability liked", "Pr(1)" = "Probability not liked"))
graphsfindbestlca.function(lcmodel4,
                           title, str.fill1, str.x1)

##Plot model selection variables
plotbicandentropy.function(LCAFilmGenresLiked$models)

##Add predicted class to SPSS file
FullSPSSData$Filmgenreclass <-
  LCAFilmGenresLiked$best$predclass

##Run and store all CHI2 results
overallchiresults <-
  setNames(data.frame(matrix(ncol = 5, nrow = 0)), c("Analysis", "chi", "df", "p",
"cv"))

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <- FullSPSSData$educTransformed
pitem <- "Education"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
```

```
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("No qualification", "GCSE", "A levels", "Degree or above")
chigenreeducation <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreEducation",
  chigenreeducation$statistic,
  chigenreeducation$parameter,
  chigenreeducation$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <- FullSPSSData$Filmwatchedclass
pitem <- "Genre"
gitem <- "Film"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("All_film",
    "Mainstrem(know)",
    "Other",
    "Mainstream(dont_know)",
    "Limited")
chigenrefilm <- chisquaretest.predictions.function(indfactor.data,
                                                   predclass.data,
                                                   length(chicols),
                                                   pitem,
                                                   gitem,
                                                   chirows,
                                                   chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreFilm",
  chigenrefilm$statistic,
  chigenrefilm$parameter,
  chigenrefilm$p.value,
  cv.test(indfactor.data, predclass.data)
)
ee
```

```
predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <- FullSPSSData$qg4_2
pitem <- "Genre"
gitem <- "Income"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Under £30K", "Over £30K")
chigenreincome <- chisquaretest.predictions.function(indfactor.data,
                                                     predclass.data,
                                                     length(chicols),
                                                     pitem,
                                                     gitem,
                                                     chirows,
                                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreIncome",
  chigenreincome$statistic,
  chigenreincome$parameter,
  chigenreincome$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <- FullSPSSData$qa2
pitem <- "Genre"
gitem <- "Gender"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Men", "Women")
chigenregender <- chisquaretest.predictions.function(indfactor.data,
                                                     predclass.data,
                                                     length(chicols),
                                                     pitem,
                                                     gitem,
                                                     chirows,
                                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreGender",
  chigenregender$statistic,
  chigenregender$parameter,
  chigenregender$p.value,
  cv.test(indfactor.data, predclass.data)
```

```
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <- FullSPSSData$qa1_2
pitem <- "Genre"
gitem <- "Age"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("15-24", "25-34", "35-54", "55+")
chigenreage <- chisquaretest.predictions.function(indfactor.data,
                                                  predclass.data,
                                                  length(chicols),
                                                  pitem,
                                                  gitem,
                                                  chirows,
                                                  chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreAge",
  chigenreage$statistic,
  chigenreage$parameter,
  chigenreage$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <- FullSPSSData$qg1
pitem <- "Genre"
gitem <- "Area"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <- c("City", "Suburb", "Town", "Village_or_rural")
chigenrearea <- chisquaretest.predictions.function(indfactor.data,
                                                   predclass.data,
                                                   length(chicols),
                                                   pitem,
                                                   gitem,
                                                   chirows,
                                                   chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreArea",
  chigenrearea$statistic,
  chigenrearea$parameter,
  chigenrearea$p.value,
```

```
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmwatchedclass
indfactor.data <- FullSPSSData$qg4_2
pitem <- "Film"
gitem <- "Income"
chicols <-
  c("All_film",
    "Mainstrem(know)",
    "Other",
    "Mainstream(dont_know)",
    "Limited")
chirows <-
  c("Under £30K", "Over £30K")
chifilmincome <- chisquaretest.predictions.function(indfactor.data,
                                                    predclass.data,
                                                    length(chicols),
                                                    pitem,
                                                    gitem,
                                                    chirows,
                                                    chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "FilmIncome",
  chifilmincome$statistic,
  chifilmincome$parameter,
  chifilmincome$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmwatchedclass
indfactor.data <- FullSPSSData$qa1_2
pitem <- "Film"
gitem <- "Age"
chicols <-
  c("All_film",
    "Mainstrem(know)",
    "Other",
    "Mainstream(dont_know)",
    "Limited")
chirows <-
  c("15-24", "25-34", "35-54", "55+")
chifilmage <- chisquaretest.predictions.function(indfactor.data,
                                                 predclass.data,
                                                 length(chicols),
                                                 pitem,
                                                 gitem,
                                                 chirows,
                                                 chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "FilmAge",
  chifilmage$statistic,
  chifilmage$parameter,
  chifilmage$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmwatchedclass
indfactor.data <- FullSPSSData$educTransformed
pitem <- "Education"
gitem <- "Class"
chicols <-
  c("All_film",
    "Mainstrem(know)",
```

```
    "Other",
    "Mainstream(dont_know)",
    "Limited")
chirows <-
  c("No qualification", "GCSE", "A levels", "Degree or above")
chifilmeducation <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "FilmEducation",
  chifilmeducation$statistic,
  chifilmeducation$parameter,
  chifilmeducation$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmwatchedclass
indfactor.data <- FullSPSSData$qa2
pitem <- "Film"
gitem <- "Gender"
chicols <-
  c("All_film",
    "Mainstrem(know)",
    "Other",
    "Mainstream(dont_know)",
    "Limited")
chirows <-
  c("Men", "Women")
chifilmgender <- chisquaretest.predictions.function(indfactor.data,
                                                    predclass.data,
                                                    length(chicols),
                                                    pitem,
                                                    gitem,
                                                    chirows,
                                                    chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "FilmGender",
  chifilmgender$statistic,
  chifilmgender$parameter,
  chifilmgender$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmwatchedclass
indfactor.data <- FullSPSSData$qg1
pitem <- "Film"
gitem <- "Area"
chicols <-
  c("All_film",
    "Mainstrem(know)",
    "Other",
    "Mainstream(dont_know)",
    "Limited")
chirows <- c("City", "Suburb", "Town", "Village_or_rural")
chifilmarea <- chisquaretest.predictions.function(indfactor.data,
                                                  predclass.data,
                                                  length(chicols),
                                                  pitem,
                                                  gitem,
```

```
                                                  chirows,
                                                  chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "FilmArea",
  chifilmarea$statistic,
  chifilmarea$parameter,
  chifilmarea$p.value,
  cv.test(indfactor.data, predclass.data)
)

##Set up perception of cultural forms scores
FullSPSSData$perceptfilmscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_1_1",
    "qb5_1_2",
    "qb5_1_3",
    "qb5_1_4",
    "qb5_1_5",
    "qb5_1_6",
    "qb5_1_7",
    "qb5_1_8",
    "qb5_1_9",
    "qb5_1_10",
    "qb5_1_11",
    "qb5_1_12",
    "qb5_1_15",
    "qb5_1_16"
  )]) - rowSums(FullSPSSData[, c("qb5_1_13", "qb5_1_14", "qb5_1_17",
                                 "qb5_1_18")]))

FullSPSSData$perceptclassicalscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_2_1",
    "qb5_2_2",
    "qb5_2_3",
    "qb5_2_4",
    "qb5_2_5",
    "qb5_2_6",
    "qb5_2_7",
    "qb5_2_8",
    "qb5_2_9",
    "qb5_2_10",
    "qb5_2_11",
    "qb5_2_12",
    "qb5_2_15",
    "qb5_2_16"
  )]) - rowSums(FullSPSSData[, c("qb5_2_13",
                                 "qb5_2_14", "qb5_2_17", "qb5_2_18")]))

FullSPSSData$perceptpopscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_3_1",
    "qb5_3_2",
    "qb5_3_3",
    "qb5_3_4",
    "qb5_3_5",
    "qb5_3_6",
    "qb5_3_7",
    "qb5_3_8",
    "qb5_3_9",
    "qb5_3_10",
    "qb5_3_11",
    "qb5_3_12",
    "qb5_3_15",
```

```
    "qb5_3_16"
  )]) - rowSums(FullSPSSData[, c("qb5_3_13",
                                 "qb5_3_14", "qb5_3_17",
                                 "qb5_3_18")]))

FullSPSSData$percepttvscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_4_1",
    "qb5_4_2",
    "qb5_4_3",
    "qb5_4_4",
    "qb5_4_5",
    "qb5_4_6",
    "qb5_4_7",
    "qb5_4_8",
    "qb5_4_9",
    "qb5_4_10",
    "qb5_4_11",
    "qb5_4_12",
    "qb5_4_15",
    "qb5_4_16"
  )]) - rowSums(FullSPSSData[, c("qb5_4_13",
                                 "qb5_4_14", "qb5_4_17",
                                 "qb5_4_18")]))


FullSPSSData$percepttheatrescore <-
  (rowSums(FullSPSSData[, c(
    "qb5_5_1",
    "qb5_5_2",
    "qb5_5_3",
    "qb5_5_4",
    "qb5_5_5",
    "qb5_5_6",
    "qb5_5_7",
    "qb5_5_8",
    "qb5_5_9",
    "qb5_5_10",
    "qb5_5_11",
    "qb5_5_12",
    "qb5_5_15",
    "qb5_5_16"
  )]) - rowSums(FullSPSSData[, c("qb5_5_13",
                                 "qb5_5_14", "qb5_5_17",
                                 "qb5_5_18")]))

FullSPSSData$perceptliteraturescore <-
  (rowSums(FullSPSSData[, c(
    "qb5_6_1",
    "qb5_6_2",
    "qb5_6_3",
    "qb5_6_4",
    "qb5_6_5",
    "qb5_6_6",
    "qb5_6_7",
    "qb5_6_8",
    "qb5_6_9",
    "qb5_6_10",
    "qb5_6_11",
    "qb5_6_12",
    "qb5_6_15",
    "qb5_6_16"
  )]) - rowSums(FullSPSSData[, c("qb5_6_13",
                                 "qb5_6_14", "qb5_6_17",
```

```
                                  "qb5_6_18")]))


FullSPSSData$perceptnewsscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_7_1",
    "qb5_7_2",
    "qb5_7_3",
    "qb5_7_4",
    "qb5_7_5",
    "qb5_7_6",
    "qb5_7_7",
    "qb5_7_8",
    "qb5_7_9",
    "qb5_7_10",
    "qb5_7_11",
    "qb5_7_12",
    "qb5_7_15",
    "qb5_7_16"
  )]) - rowSums(FullSPSSData[, c("qb5_7_13",
                                  "qb5_7_14",    "qb5_7_17",
                                  "qb5_7_18")]))

FullSPSSData$perceptartscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_8_1",
    "qb5_8_2",
    "qb5_8_3",
    "qb5_8_4",
    "qb5_8_5",
    "qb5_8_6",
    "qb5_8_7",
    "qb5_8_8",
    "qb5_8_9",
    "qb5_8_10",
    "qb5_8_11",
    "qb5_8_12",
    "qb5_8_15",
    "qb5_8_16"
  )]) - rowSums(FullSPSSData[, c("qb5_8_13",
                                  "qb5_8_14", "qb5_8_17",
                                  "qb5_8_18")]))

FullSPSSData$perceptgamesscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_9_1",
    "qb5_9_2",
    "qb5_9_3",
    "qb5_9_4",
    "qb5_9_5",
    "qb5_9_6",
    "qb5_9_7",
    "qb5_9_8",
    "qb5_9_9",
    "qb5_9_10",
    "qb5_9_11",
    "qb5_9_12",
    "qb5_9_15",
    "qb5_9_16"
  )]) - rowSums(FullSPSSData[, c("qb5_9_13", "qb5_9_14", "qb5_9_17",
"qb5_9_18")]))

FullSPSSData$perceptsportscore <-
  (rowSums(FullSPSSData[, c(
```

```
    "qb5_10_1",
    "qb5_10_2",
    "qb5_10_3",
    "qb5_10_4",
    "qb5_10_5",
    "qb5_10_6",
    "qb5_10_7",
    "qb5_10_8",
    "qb5_10_9",
    "qb5_10_10",
    "qb5_10_11",
    "qb5_10_12",
    "qb5_10_15",
    "qb5_10_16"
  )])  - rowSums(FullSPSSData[, c("qb5_10_13", "qb5_10_14", "qb5_10_17",
"qb5_10_18")]))


FullSPSSData$perceptsportpartscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_11_1",
    "qb5_11_2",
    "qb5_11_3",
    "qb5_11_4",
    "qb5_11_5",
    "qb5_11_6",
    "qb5_11_7",
    "qb5_11_8",
    "qb5_11_9",
    "qb5_11_10",
    "qb5_11_11",
    "qb5_11_12",
    "qb5_11_15",
    "qb5_11_16"
  )]) - rowSums(FullSPSSData[, c("qb5_11_13",
                                 "qb5_11_14", "qb5_11_17",
                                 "qb5_11_18")]))

FullSPSSData$perceptpubsscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_12_1",
    "qb5_12_2",
    "qb5_12_3",
    "qb5_12_4",
    "qb5_12_5",
    "qb5_12_6",
    "qb5_12_7",
    "qb5_12_8",
    "qb5_12_9",
    "qb5_12_10",
    "qb5_12_11",
    "qb5_12_12",
    "qb5_12_15",
    "qb5_12_16"
  )]) - rowSums(FullSPSSData[, c("qb5_12_13",
                                 "qb5_12_14", "qb5_12_17",
                                 "qb5_12_18")]))

FullSPSSData$perceptrestscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_13_1",
    "qb5_13_2",
    "qb5_13_3",
    "qb5_13_4",
```

```
    "qb5_13_5",
    "qb5_13_6",
    "qb5_13_7",
    "qb5_13_8",
    "qb5_13_9",
    "qb5_13_10",
    "qb5_13_11",
    "qb5_13_12",
    "qb5_13_15",
    "qb5_13_16"
  )])  - rowSums(FullSPSSData[, c("qb5_13_13",
                                  "qb5_13_14", "qb5_13_17",
                                  "qb5_13_18")]))

FullSPSSData$perceptcountryscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_14_1",
    "qb5_14_2",
    "qb5_14_3",
    "qb5_14_4",
    "qb5_14_5",
    "qb5_14_6",
    "qb5_14_7",
    "qb5_14_8",
    "qb5_14_9",
    "qb5_14_10",
    "qb5_14_11",
    "qb5_14_12",
    "qb5_14_15",
    "qb5_14_16"
  )])  - rowSums(FullSPSSData[, c("qb5_14_13",
                                  "qb5_14_14", "qb5_14_17",
                                  "qb5_14_18")]))


FullSPSSData$perceptreligionscore <-
  (rowSums(FullSPSSData[, c(
    "qb5_15_1",
    "qb5_15_2",
    "qb5_15_3",
    "qb5_15_4",
    "qb5_15_5",
    "qb5_15_6",
    "qb5_15_7",
    "qb5_15_8",
    "qb5_15_9",
    "qb5_15_10",
    "qb5_15_11",
    "qb5_15_12",
    "qb5_15_15",
    "qb5_15_16"
  )])     - rowSums(FullSPSSData[, c("qb5_15_13",
                                     "qb5_15_14", "qb5_15_17",
                                     "qb5_15_18")]))

FullSPSSData$filmactivities <-
  rowSums(FullSPSSData[, c("qc7_1",
                           "qc7_2",
                           "qc7_3",
                           "qc7_4",
                           "qc7_5")], na.rm = TRUE)

##Bin scores into 4 levels
FullSPSSData$perceptfilmscorebin <-
```

```
  binVariable(FullSPSSData$perceptfilmscore, bins = 4, "natural")
FullSPSSData$perceptclassicalscorebin <-
  binVariable(FullSPSSData$perceptclassicalscore, bins = 4, "natural")
FullSPSSData$perceptpopscorebin <-
  binVariable(FullSPSSData$perceptpopscore, bins = 4, "natural")
FullSPSSData$percepttvscorebin <-
  binVariable(FullSPSSData$percepttvscore, bins = 4, "natural")
FullSPSSData$perceptheatrescorebin <-
  binVariable(FullSPSSData$perceptheatrescore, bins = 4, "natural")
FullSPSSData$perceptliteraturescorebin <-
  binVariable(FullSPSSData$perceptliteraturescore, bins = 4, "natural")
FullSPSSData$perceptartscorebin <-
  binVariable(FullSPSSData$perceptartscore, bins = 4, "natural")
FullSPSSData$perceptgamesscorebin <-
  binVariable(FullSPSSData$perceptgamesscore, bins = 4, "natural")
FullSPSSData$perceptsportscorebin <-
  binVariable(FullSPSSData$perceptsportscore, bins = 4, "natural")
FullSPSSData$perceptsportpartscorebin <-
  binVariable(FullSPSSData$perceptsportpartscore, bins = 4, "natural")
FullSPSSData$perceptpubsscorebin <-
  binVariable(FullSPSSData$perceptpubsscore, bins = 4, "natural")
FullSPSSData$perceptrestscorebin <-
  binVariable(FullSPSSData$perceptrestscore, bins = 4, "natural")
FullSPSSData$perceptcountryscorebin <-
  binVariable(FullSPSSData$perceptcountryscore, bins = 4, "natural")
FullSPSSData$perceptreligionscorebin <-
  binVariable(FullSPSSData$perceptreligionscore, bins = 4, "natural")


##Run CHI2 on perceptions
predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptfilmscore, bins = 4, "natural")
FullSPSSData$perceptfilmscorebin <- indfactor.data
pitem <- "Perception of Film"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenrefilmP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreFilmP",
  chigenrefilmP$statistic,
  chigenrefilmP$parameter,
  chigenrefilmP$p.value,
  cv.test(indfactor.data, predclass.data)
```

```
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptclassicalscore, bins = 4, "natural")
FullSPSSData$perceptclassicalscorebin <- indfactor.data
pitem <- "Perception of Classical Music"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenreclassicalP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreClassicalP",
  chigenreclassicalP$statistic,
  chigenreclassicalP$parameter,
  chigenreclassicalP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptpopscore, bins = 4, "natural")
FullSPSSData$perceptpopscorebin <- indfactor.data
pitem <- "Perception of Pop"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenrepopP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
```

```
                                    chirows,
                                    chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenrePopP",
  chigenrepopP$statistic,
  chigenrepopP$parameter,
  chigenrepopP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$percepttvscore, bins = 4, "natural")
FullSPSSData$percepttvscorebin <- indfactor.data
pitem <- "Perception of TV"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenretvP <-
  chisquaretest.predictions.function(indfactor.data,
                                    predclass.data,
                                    length(chicols),
                                    pitem,
                                    gitem,
                                    chirows,
                                    chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreTVP",
  chigenretvP$statistic,
  chigenretvP$parameter,
  chigenretvP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$percepttheatrescore, bins = 4, "natural")
FullSPSSData$percepttheatrescorebin <- indfactor.data
pitem <- "Perception of Theatre"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
```

```
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenretheatreP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreTheatreP",
  chigenretheatreP$statistic,
  chigenretheatreP$parameter,
  chigenretheatreP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptliteraturescore, bins = 4, "natural")
pitem <- "Perception of Literature"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenrelitP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreLitP",
  chigenrelitP$statistic,
  chigenrelitP$parameter,
  chigenrelitP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptnewsscore, bins = 4, "natural")
FullSPSSData$perceptnewsscorebin <- indfactor.data
pitem <- "Perception of News"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
```

```
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenrenewsP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreNewsP",
  chigenrenewsP$statistic,
  chigenrenewsP$parameter,
  chigenrenewsP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptartscore, bins = 4, "natural")
FullSPSSData$perceptartscorebin <- indfactor.data
pitem <- "Perception of Arts"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenreartsP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreArtsP",
  chigenreartsP$statistic,
  chigenreartsP$parameter,
  chigenreartsP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptgamesscore, bins = 4, "natural")
```

```
FullSPSSData$perceptgamesscorebin <- indfactor.data
pitem <- "Perception of Video Games"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenregamesP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreGamesP",
  chigenregamesP$statistic,
  chigenregamesP$parameter,
  chigenregamesP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptsportscore, bins = 4, "natural")
FullSPSSData$perceptsportscorebin <- indfactor.data
pitem <- "Perception of Sports Attendance"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenresportsAP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreSportsAP",
  chigenresportsAP$statistic,
```

```
  chigenresportsAP$parameter,
  chigenresportsAP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptsportpartscore, bins = 4, "natural")
FullSPSSData$perceptsportpartscorebin <- indfactor.data
pitem <- "Perception of Sports Participation"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenresportsPP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreSportsPP",
  chigenresportsPP$statistic,
  chigenresportsPP$parameter,
  chigenresportsPP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptpubsscore, bins = 4, "natural")
FullSPSSData$perceptpubsscorebin <- indfactor.data
pitem <- "Perception of Pubs and Clubs"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenrespubsP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
```

```
                                    length(chicols),
                                    pitem,
                                    gitem,
                                    chirows,
                                    chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenrePubsP",
  chigenrespubsP$statistic,
  chigenrespubsP$parameter,
  chigenrespubsP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptrestscore, bins = 4, "natural")
FullSPSSData$perceptrestscorebin <- indfactor.data
pitem <- "Perception of Restaurants"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenresrestP <-
  chisquaretest.predictions.function(indfactor.data,
                                    predclass.data,
                                    length(chicols),
                                    pitem,
                                    gitem,
                                    chirows,
                                    chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreCRestP",
  chigenresrestP$statistic,
  chigenresrestP$parameter,
  chigenresrestP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptcountryscore, bins = 4, "natural")
FullSPSSData$perceptcountryscorebin <- indfactor.data
pitem <- "Perception of Countryside"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
```

```
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenrescountryP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreCountryP",
  chigenrescountryP$statistic,
  chigenrescountryP$parameter,
  chigenrescountryP$p.value,
  cv.test(indfactor.data, predclass.data)
)

predclass.data <- FullSPSSData$Filmgenreclass
indfactor.data <-
  binVariable(FullSPSSData$perceptreligionscore, bins = 4, "natural")
FullSPSSData$perceptreligionscorebin <- indfactor.data
pitem <- "Perception of Religion"
gitem <- "Genre"
chicols <-
  c(
    "Suspense",
    "Drama",
    "Romantic",
    "Comedy",
    "Main_stream",
    "Family",
    "SF_fantasy",
    "Art_house_foreign",
    "No_preference"
  )
chirows <-
  c("Negative", "Limited", "Postive", "Very positive")
chigenresrelP <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreRelP",
  chigenresrelP$statistic,
  chigenresrelP$parameter,
  chigenresrelP$p.value,
  cv.test(indfactor.data, predclass.data)
)

FullSPSSData$Specialised <- 0
FullSPSSData$Specialised[FullSPSSData$Filmgenreclass == 8] <- 1
FullSPSSData$Watchedspecial <- 0
FullSPSSData$Watchedspecial[(
  FullSPSSData$qc3a_3 +
    FullSPSSData$qc3a_4 +
    FullSPSSData$qc3b_3 +
```

```
    FullSPSSData$qc3c_4 +
    FullSPSSData$qc3d_3 +
    FullSPSSData$qc3d_4 +
    FullSPSSData$qc3e_3 +
    FullSPSSData$qc3e_4 +
    FullSPSSData$qc3f_3 +
    FullSPSSData$qc3f_4
) > 0] <- 1
FullSPSSData$Likeandwatch <- 0
FullSPSSData$Likeandwatch[(FullSPSSData$Specialised + FullSPSSData$Watchedspecial)
> 0] <-
  1

predclass.data <- FullSPSSData$Watchedspecial
indfactor.data <- FullSPSSData$ClusterArtHouse
FullSPSSData$perceptreligionscorebin <- indfactor.data
pitem <- "Watched special"
gitem <- "Like art house or foreign language"
chicols <-
  c("Did not watch", "Did watch")
chirows <-
  c("Does not like", "Likes")
chiwatchandlike <-
  chisquaretest.predictions.function(indfactor.data,
                                     predclass.data,
                                     length(chicols),
                                     pitem,
                                     gitem,
                                     chirows,
                                     chicols)
overallchiresults[nrow(overallchiresults) + 1, ] = list(
  "GenreRelP",
  chiwatchandlike$statistic,
  chiwatchandlike$parameter,
  chiwatchandlike$p.value,
  cv.test(indfactor.data, predclass.data)
)

##SAve CHI2 results
savefile <- file.path(getwd(), "chiresults.csv")
write.csv(overallchiresults, savefile)

spssdatafile <- file.path(getwd(), "FilmanalysisRFinal.sav")
write_sav(FullSPSSData, spssdatafile)

##Start and set up binary logistic regressions
sink(file = "glmresults.txt",
     append = FALSE,
     split = TRUE)

FullSPSSData$qa1_1[is.na(FullSPSSData$qa1_1)] <- 1
FullSPSSData$qa2[is.na(FullSPSSData$qa2)] <- 1
FullSPSSData$qg1[is.na(FullSPSSData$qg1)] <- 1
FullSPSSData$qg4_2[is.na(FullSPSSData$qg4_2)] <- 1
FullSPSSData$educTransformed[is.na(FullSPSSData$educTransformed)] <-
  1
FullSPSSData$allfilm <- 0
FullSPSSData$allfilm[FullSPSSData$Filmwatchedclass == 1] <- 1
FullSPSSData$filmactivities <- sum()

fit.logistf.specialised <- logistf(
  Specialised ~
    perceptfilmscore +
    perceptclassicalscore +
```

```
    perceptpopscore +
    percepttvscore +
    perceptheatrescore +
    perceptliteraturescore +
    perceptnewsscore +
    perceptartscore +
    perceptgamesscore +
    perceptsportscore +
    perceptsportpartscore +
    perceptpubsscore +
    perceptrestscore +
    perceptcountryscore +
    perceptreligionscore +
    educTransformed +
    qa1_1 +
    qa2 +
    qg1 +
    qg4_2,
  data = FullSPSSData
)

FullSPSSData$modelspecialised05 <- 0
FullSPSSData$modelspecialised05[fit.logistf.specialised$predict > 0.5] <-
  1
FullSPSSData$modelspecialised03 <- 0
FullSPSSData$modelspecialised03[fit.logistf.specialised$predict > 0.3] <-
  1
FullSPSSData$modellinearspecialised <- 0
FullSPSSData$modellinearspecialised[fit.logistf.specialised$linear.predict >
                                      0] <- 1

# 2-Way Cross Tabulation
CrossTable(FullSPSSData$modelspecialised05, FullSPSSData$Specialised)
CrossTable(FullSPSSData$modelspecialised03, FullSPSSData$Specialised)
CrossTable(FullSPSSData$modellinearspecialised,
           FullSPSSData$Specialised)
CrossTable(FullSPSSData$modelspecialised05,
           FullSPSSData$modellinearspecialised)

fit.glm.likeandwatchbin <- glm(
  Likeandwatch ~
    perceptfilmscorebin +
    perceptclassicalscorebin +
    perceptpopscorebin +
    percepttvscorebin +
    perceptheatrescorebin +
    perceptliteraturescorebin +
    perceptnewsscorebin +
    perceptartscorebin +
    perceptgamesscorebin +
    perceptsportscorebin +
    perceptsportpartscorebin +
    perceptpubsscorebin +
    perceptrestscorebin +
    perceptcountryscorebin +
    perceptreligionscorebin +
    educTransformed +
    qa1_1 +
    qa2 +
    qg1 +
    qg4_2,
  family = binomial(link = "logit"),
  data = FullSPSSData
)
```

```
summary(fit.glm.likeandwatchbin)
or.fit.glm.likeandwatchbin <- exp(cbind(
  OR = coef(fit.glm.likeandwatchbin),
  confint(fit.glm.likeandwatchbin)
))
savefile <- file.path(getwd(), "summaryfitglmlikeandwatchbin.csv")
write.csv(summary.fit.glm.likeandwatchbin, savefile)

savefile <- file.path(getwd(), "orfitglmlikeandwatchbin.csv")
write.csv(or.fit.glm.likeandwatchbin, savefile)

FullSPSSData$modelLikeandwatchbin05 <- 0
FullSPSSData$modelLikeandwatchbin05[fit.glm.likeandwatchbin$predict > 0.5] <-
  1
FullSPSSData$modelLikeandwatchbin03 <- 0
FullSPSSData$modelLikeandwatchbin03[fit.glm.likeandwatchbin$predict > 0.3] <-
  1
FullSPSSData$modellinearLikeandwatchbin <- 0
FullSPSSData$modellinearLikeandwatchbin[fit.glm.likeandwatchbin$linear.predict >
                                          0] <- 1

##Accuracy
accuracy <-
  table(FullSPSSData$modelLikeandwatchbin05,
        FullSPSSData$Likeandwatch)
sum(diag(accuracy)) / sum(accuracy)

fit.glmlogit.likedandwatched <- glm(
  Likeandwatch ~
    perceptfilmscore +
    perceptclassicalscore +
    perceptpopscore +
    percepttvscore +
    perceptheatrescore +
    perceptliteraturescore +
    perceptnewsscore +
    perceptartscore +
    perceptgamesscore +
    perceptsportscore +
    perceptsportpartscore +
    perceptpubsscore +
    perceptrestscore +
    perceptcountryscore +
    perceptreligionscore +
    educTransformed +
    qa1_1 +
    qa2 +
    qg1 +
    qg4_2,
  family = binomial(link = "logit"),
  data = FullSPSSData
)
FullSPSSData$modelLikeandwatchglm05 <- 0
FullSPSSData$modelLikeandwatchglm05[fit.glmlogit.likedandwatched$predict > 0.5] <-
  1
FullSPSSData$modelLikeandwatchglm03 <- 0
FullSPSSData$modelLikeandwatchglm03[fit.glmlogit.likedandwatched$predict > 0.3] <-
  1
FullSPSSData$modellinearLikeandwatchglm <- 0
FullSPSSData$modellinearLikeandwatchglm[fit.glmlogit.likedandwatched$linear.predict
 >
                                          0] <- 1

##Accuracy
```

```
accuracy <-
  table(FullSPSSData$modelLikeandwatchglm05,
        FullSPSSData$Likeandwatch)
sum(diag(accuracy)) / sum(accuracy)

summary(fit.glm.likeandwatch)
PseudoR2(fit.glmlogit.likedandwatched, which = "Nagelkerke")
PseudoR2(fit.glmlogit.likedandwatched, which = "CoxSnell")
PseudoR2(fit.glmlogit.likedandwatched, which = "Tjur")
logitgof(fit.glmlogit.likedandwatched$y,
         fitted(fit.glmlogit.likedandwatched),
         g = 10)

fit.glmlogit.likedandwatched0 <- glm(Likeandwatch ~ 1,
                                     family = binomial(link = "logit"),
                                     data = FullSPSSData)
or.fit.glm.likeandwatch <- exp(cbind(
  coef(fit.glmlogit.likedandwatched),
  confint(fit.glmlogit.likedandwatched)
))
savefile <- file.path(getwd(), "orfitglmlikeandwatch.csv")
write.csv(or.fit.glm.likeandwatch, savefile)

anova(fit.glmlogit.likedandwatched0,
      fit.glmlogit.likedandwatched,
      test = "Chisq")

fit.glmlogit.like <- glm(
  ClusterArtHouse ~
    perceptfilmscore +
    perceptclassicalscore +
    perceptpopscore +
    perceptttvscore +
    perceptttheatrescore +
    perceptliteraturescore +
    perceptnewsscore +
    perceptartscore +
    perceptgamesscore +
    perceptsportscore +
    perceptsportpartscore +
    perceptpubsscore +
    perceptrestscore +
    perceptcountryscore +
    perceptreligionscore +
    educTransformed +
    qa1_1 +
    qa2 +
    qg1 +
    qg4_2,
  family = binomial(link = "logit"),
  data = FullSPSSData
)
FullSPSSData$modellikeglm05 <- 0
FullSPSSData$modellikeglm05[fit.glmlogit.like$predict > 0.5] <-
  1
FullSPSSData$modellikeglm03 <- 0
FullSPSSData$modellikeglm03[fit.glmlogit.like$predict > 0.3] <-
  1
FullSPSSData$modellinearlikeglm <- 0
FullSPSSData$modellinearlikeglm[fit.glmlogit.like$linear.predict >
                                  0] <- 1

##Accuracy
accuracy <-
```

```
  table(FullSPSSData$modellikeglm05, FullSPSSData$ClusterArtHouse)
sum(diag(accuracy)) / sum(accuracy)

summary(fit.glmlogit.like)
PseudoR2(fit.glmlogit.like, which = "Nagelkerke")
PseudoR2(fit.glmlogit.like, which = "CoxSnell")
logitgof(fit.glmlogit.like$y,
         fitted(fit.glmlogit.like),
         g = 10)


fit.glmlogit.like0 <- glm(ClusterArtHouse ~ 1,
                          family = binomial(link = "logit"),
                          data = FullSPSSData)
exp(cbind(coef(fit.glmlogit.like),
          confint(fit.glmlogit.like)))
anova(fit.glmlogit.like0,
      fit.glmlogit.like,
      test = "Chisq")

fit.glmlogit.allfilm <- glm(
  allfilm ~
    perceptfilmscore +
    perceptclassicalscore +
    perceptpopscore +
    percepttvscore +
    percepttheatrescore +
    perceptliteraturescore +
    perceptnewsscore +
    perceptartscore +
    perceptgamesscore +
    perceptsportscore +
    perceptsportpartscore +
    perceptpubsscore +
    perceptrestscore +
    perceptcountryscore +
    perceptreligionscore +
    educTransformed +
    qa1_1 +
    qa2 +
    qg1 +
    qg4_2,
  family = binomial(link = "logit"),
  data = FullSPSSData
)

summary(fit.glmlogit.allfilm)
PseudoR2(fit.glmlogit.allfilm, which = "Nagelkerke")
PseudoR2(fit.glmlogit.allfilm, which = "CoxSnell")
logitgof(fit.glmlogit.allfilm$y,
         fitted(fit.glmlogit.allfilm),
         g = 10)

fit.glmlogit.allfilm0 <- glm(allfilm ~ 1,
                             family = binomial(link = "logit"), data =
FullSPSSData)
exp(cbind(coef(fit.glmlogit.allfilm),
          confint(fit.glmlogit.allfilm)))
anova(fit.glmlogit.allfilm0,
      fit.glmlogit.allfilm,
      test = "Chisq")

FullSPSSData$modelglmlinear <- 0
FullSPSSData$modelglmlinear[fit.glmlogit.allfilm$linear.predict > 0] <-
```

```
  1
accuracy <- table(FullSPSSData$allfilm, FullSPSSData$modelglmlinear)
sum(diag(accuracy)) / sum(accuracy)


fit.glmlogit.allfilmbin <- glm(
  allfilm ~
    perceptfilmscorebin +
    perceptclassicalscorebin +
    perceptpopscorebin +
    percepttvscorebin +
    perceptheatrescorebin +
    perceptliteraturescorebin +
    perceptnewsscorebin +
    perceptartscorebin +
    perceptgamesscorebin +
    perceptsportscorebin +
    perceptsportpartscorebin +
    perceptpubsscorebin +
    perceptrestscorebin +
    perceptcountryscorebin +
    perceptreligionscorebin +
    educTransformed +
    qa1_1 +
    qa2 +
    qg1 +
    qg4_2,
  family = binomial(link = "logit"),
  data = FullSPSSData
)

summary(fit.glmlogit.allfilmbin)
PseudoR2(fit.glmlogit.allfilmbin, which = "Nagelkerke")
PseudoR2(fit.glmlogit.allfilmbin, which = "CoxSnell")
logitgof(fit.glmlogit.allfilmbin$y,
         fitted(fit.glmlogit.allfilmbin),
         g = 10)

fit.glmlogit.allfilmbin0 <- glm(allfilm ~ 1,
                                family = binomial(link = "logit"), data =
FullSPSSData)
exp(cbind(
  coef(fit.glmlogit.allfilmbin),
  confint(fit.glmlogit.allfilmbin)
))
anova(fit.glmlogit.allfilmbin0,
      fit.glmlogit.allfilmbin,
      test = "Chisq")

FullSPSSData$modelglmlinearbin <- 0
FullSPSSData$modelglmlinearbin[fit.glmlogit.allfilmbin$linear.predict > 0] <-
  1
accuracy <- table(FullSPSSData$allfilm, FullSPSSData$modelglmlinearbin)
sum(diag(accuracy)) / sum(accuracy)

fit.glmlogit.allfilmsimple <- glm(
  allfilm ~
    educTransformed +
    qa1_1 +
    qa2 +
    qg1 +
    qg4_2,
  family = binomial(link = "logit"),
  data = FullSPSSData
```

```
)

summary(fit.glmlogit.allfilmsimple)
PseudoR2(fit.glmlogit.allfilmsimple, which = "Nagelkerke")
PseudoR2(fit.glmlogit.allfilmsimple, which = "CoxSnell")
logitgof(fit.glmlogit.allfilmsimple$y,
         fitted(fit.glmlogit.allfilmsimple),
         g = 8)

fit.glmlogit.allfilmsimple0 <- glm(allfilm ~ 1,
                                   family = binomial(link = "logit"), data =
FullSPSSData)
exp(cbind(
  coef(fit.glmlogit.allfilmsimple),
  confint(fit.glmlogit.allfilmsimple)
))
anova(fit.glmlogit.allfilmsimple0,
      fit.glmlogit.allfilmsimple,
      test = "Chisq")

anova(fit.glmlogit.allfilmsimple,
      fit.glmlogit.allfilm,
      test = "Chisq")

anova(fit.glmlogit.allfilmsimple,
      fit.glmlogit.allfilmbin,
      test = "Chisq")

FullSPSSData$modelglmlinearsimple <- 0
FullSPSSData$modelglmlinearsimple[fit.glmlogit.allfilmsimple$linear.predict >
                                    0] <- 1
accuracy <-
  table(FullSPSSData$allfilm, FullSPSSData$modelglmlinearsimple)
sum(diag(accuracy)) / sum(accuracy)

sink()

#dev.off()
```