

Technical Appendix

CK

22/11/2018

```
# Import ATUS Summary file
ATUS.SUM <- import('atussum_0317.csv', setclass = "tibble")
# Import ATUS CPS File
ATUS.CPS <- import('atuscps_0317.csv', setclass = "tibble")
# Create a diary date column
ATUS.SUM$TUDIARYDATE <- as.Date(substr(ATUS.SUM$TUCASEID, 1, 8), "%Y%m%d")
# Add months as a column as a derivative of the TUCASEID variable
ATUS.SUM$month <- as.numeric(substr(ATUS.SUM$TUCASEID, 5, 6))
# Filter CPS down to just relevant variables. GEDSTFIPS is state, GEDIV is division and GEREG is region
ATUS.CPS.States <- select(ATUS.CPS, TUCASEID, GESTFIPS, GEDIV, GEREG)
# Create variables to be added to a data frame with relevant gender activities
Working <- ATUS.SUM[, grep("^t05", names(ATUS.SUM))]
House.Maintenance <- ATUS.SUM[, grep("^t0204", names(ATUS.SUM))]
Vehicle.Maintenance <- ATUS.SUM[, grep("^t0207", names(ATUS.SUM))]
Housework <- ATUS.SUM[, grep("^t0201", names(ATUS.SUM))]
Food.Prep <- ATUS.SUM[, grep("^t0202", names(ATUS.SUM))] + ATUS.SUM$t070101 + ATUS.SUM$t070103
Childcare <- ATUS.SUM[, grep("^t03", names(ATUS.SUM))]

# Create a data frame from relevant variables within ATUS Summary
# Childnum there for future use
Gender.Roles.Data <- select(ATUS.SUM, TUCASEID, TESEX, TEAGE, TUFNWGTP, TUYEAR, TRCHILDNUM, month, TUDIARYDATE)
# Join this dataset with the CPS states info by unique id number of participant
# Add in relevant categories
Gender.Roles.Data$Working <- rowSums(Working)
Gender.Roles.Data$House.Maintenance <- rowSums(House.Maintenance)
Gender.Roles.Data$Vehicle.Maintenance <- rowSums(Vehicle.Maintenance)
Gender.Roles.Data$Housework <- rowSums(Housework)
Gender.Roles.Data$Food.Prep <- rowSums(Food.Prep)
Gender.Roles.Data$Childcare <- rowSums(select(Childcare, t030101:t030399))
# Join this dataset with the CPS states info by unique id number of participant
Gender.Roles.Data <- left_join(Gender.Roles.Data, ATUS.CPS.States, by = 'TUCASEID')
Gender.Roles.Data <- filter(Gender.Roles.Data)
Gender.Roles.Data$TESEX[Gender.Roles.Data$TESEX == 1] <- "M"
Gender.Roles.Data$TESEX[Gender.Roles.Data$TESEX == 2] <- "F"

for (variable in c("Working", "House.Maintenance", "Vehicle.Maintenance", "Housework", "Food.Prep", "Childcare")) {
  Gender.Roles.Data[, paste0('weighted.', variable)] <- 0
  for (sex in c("M", "F")) {
    for (year in unique(Gender.Roles.Data$TUYEAR)) {
      for (month in unique(Gender.Roles.Data$month)) {
        Filtered.Data <- filter(Gender.Roles.Data, TESEX == sex, TUYEAR == year, month == month)
        bottom.sum <- sum(Filtered.Data$TUFNWGTP)
        top.sum <- sum(Filtered.Data$TUFNWGTP * Filtered.Data[, variable])
        weighted.values <- top.sum / bottom.sum

        Gender.Roles.Data[Gender.Roles.Data$TESEX == sex & Gender.Roles.Data$TUYEAR == year & Gender.Roles.Data$month == month,
          paste0('weighted.', variable)] <- weighted.values
      }
    }
  }
}
```

```

    }
  }
  Gender.Roles.Train <- filter(Gender.Roles.Data, month %% 2 == 0)
  Gender.Roles.Validate <- filter(Gender.Roles.Data, month %% 2 != 0)

  for (variable in c("Working", "House.Maintenance", "Vehicle.Maintenance", "Housework", "Food.Prep", "Childcare")) {
    # Start building models for the data
    n <- nrow(Gender.Roles.Train)
    model1 <- glm(as.formula(paste0('weighted.', variable, ' ~ ns(TUDIARYDATE, knots = (c(as.Date("2006-01-01"), as.Date("2006-01-01"))))), data = Gender.Roles.Train)
    model2 <- update(model1, . ~ . + TESEX)
    model3 <- update(model1, . ~ -1 + TESEX + TESEX:ns(TUDIARYDATE, knots = (c(as.Date('2006-01-01'), as.Date('2006-01-01'))))), data = Gender.Roles.Train)
    # Use same method as in lab 5 to plot them
    male.prediction <- predict.glm(model3, newdata = filter(Gender.Roles.Train, TESEX == 'M'), se = TRUE)
    female.prediction <- predict.glm(model3, newdata = filter(Gender.Roles.Train, TESEX == 'F'), se = TRUE)

    df <- data.frame(age = c(filter(Gender.Roles.Train, TESEX == 'M')$TEAGE, filter(Gender.Roles.Train, TESEX == 'F')$TEAGE),
                     prediction = c(male.prediction, female.prediction),
                     prediction_error = c(male.prediction_error, female.prediction_error))

    model3_plot <- ggplot(df, aes(x=date, y=prediction, colour=type)) +
      geom_line() +
      geom_ribbon(aes(ymin = prediction - prediction_error, ymax = prediction + prediction_error), alpha = 0.5)
    model3_plot
  }
}

```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: algorithm did not converge
```