Technical Appendix

Group July

26 November 2018

```
# - Ages from 80-84 are rounded to 80 and those 85+ are rounded to 85
# - In the data -1 means blank, -2 means don't know, -3 means refused
```

Preparing the Environment

(Please note that the `tidy` function in R-Markdown does not work correctly, and usually destroys the indentation / formatting of code which can lead to it being unreadable especially in code blocks as long as these, it is suggested for the reader's sanity that they look at the code in the `.rmd` file rather than the pdf but this choice can only be left to the reader)

First, load all of the required packages and install any that are missing.

```
if (!require("kableExtra")){
  install.packages("kableExtra")
  library("kableExtra")
} # To change kable options (https://www.rdocumentation.org/packages/kableExtra/versions/0.9.0)
if (!require("dplyr")){
  install.packages("dplyr")
  library("dplyr")
} # data manipulation (https://www.rdocumentation.org/packages/dplyr/versions/0.7.8)
if (!require("rio")){
  install.packages("rio")
  library("rio")
} # Allows us to use tibbles (https://www.rdocumentation.org/packages/rio/versions/0.5.10)
if (!require("zoo")){
  install.packages("zoo")
  library("zoo")
} # Allows us to use dates (https://www.rdocumentation.org/packages/zoo/versions/1.8-4)
if (!require("lubridate")){
  install.packages("lubridate")
  library("lubridate")
} # Allows us to use the function 'month' (https://www.rdocumentation.org/packages/lubridate/vers
if (!require("reshape2")){
  install.packages("reshape2")
  library("reshape2")
} # Allows us to use the function 'melt' (https://www.rdocumentation.org/packages/reshape2/versio
if (!require("ggplot2")){
  install.packages("ggplot2")
  library("ggplot2")
} # More control over plots (https://www.rdocumentation.org/packages/ggplot2/versions/3.1.0)
if (!require("knitr")){
  install.packages("knitr")
  library("knitr")
```

```
} # To create tables using kable (https://www.rdocumentation.org/packages/knitr/versions/1.20)
if (!require("splines")){
  install.packages("splines")
  library("splines")
} # To use spline curves for models (https://www.rdocumentation.org/packages/splines/versions/3.5
if (!require("leaps")){
  install.packages("leaps")
  library("leaps")
} # To use best subsets for models (https://www.rdocumentation.org/packages/leaps/versions/3.0)
if (!require("modEvA")){
  install.packages("modEvA")
  library("modEvA")
} # To calculate D-Squared (https://www.rdocumentation.org/packages/modEvA/versions/1.3.2)
if (!require("geojsonio")){
  install.packages("geojsonio")
  library("geojsonio")
} # To read a .geojson file into R to different types for map use (https://www.rdocumentation.org
if (!require("openintro")){
  install.packages("openintro")
  library("openintro")
} # To convert state abbreviations to state names (https://www.rdocumentation.org/packages/openin
if (!require("broom")){
  install.packages("broom")
  library("broom")
} # To tidy and fortify the data into a data frame for map use (https://www.rdocumentation.org/pa
if (!require("rgeos")){
  install.packages("rgeos")
  library("rgeos")
} # To calculate the centre of each state for map use (https://www.rdocumentation.org/packages/rg
```

2 key conclusions are mentioned in the report. The code below shows the methods used to reach these conclusions. Both follow the structure of Exploratory Analysis, Validation and then a final plot using all the data, excluding July.

Loading and Tidying the Data

Import the datasets required from the ATUS [1] corpus of data. Summary and CPS files are used primarily.

```
ATUS_dataset <- import("atussum_0317.csv", setclass = "tibble")    # import raw summary data as T
ATUS.CPS <- import('atuscps_0317.csv', setclass = "tibble") # import CPS data as TIBBLE for geogr
```

Now some tidying and subsetting of the data can take place for use in the first section of the report.

```
# Check there are no other NA's other than those in format explained in the notes
NA_check <- sum(is.na(ATUS_dataset))

# Add date from respondent file and change Sex values to M and F instead of 1 and 2
ATUS_dataset <- mutate(ATUS_dataset, TESEX = ifelse(TESEX == 1, "M", "F"))

# Create a diary date column based off of the unique ID
```

```r
ATUS_dataset$TUDIARYDATE <- as.Date(paste0(substr(ATUS_dataset$TUCASEID, 1, 6), '01'), "%Y%m%d")
ATUS_dataset_edit <- ATUS_dataset

# A change of variable name in this period leads to missing values so we merge these 2 variables
ATUS_dataset_edit <- mutate(ATUS_dataset_edit, Met_Status = as.factor(pmax(GEMETSTA, GTMETSTA)))
ATUS_dataset_edit <- select(ATUS_dataset_edit, -one_of(c("GEMETSTA", "GTMETSTA")))

# Set suitable variables as factors
factor_columns <- c("PEEDUCA", "PEHSPNON", "PTDTRACE", "TELFS", "TEMJOT", "TESCHENR", "TESCHLVL",
ATUS_dataset_edit[,factor_columns] <- lapply(ATUS_dataset_edit[,factor_columns], factor)

# Create time spent and participation columns with the 17 Categories
for (i in 1:9){
  assign(paste0("tu0", i), ATUS_dataset_edit[,grep(paste0("^t0", i), names(ATUS_dataset_edit))])
  assign(paste0("tu0",i), rowSums(get(paste0("tu0",i))))
  ATUS_dataset_edit <- bind_cols(ATUS_dataset_edit, tu0 = get(paste0("tu0", i)))
  names(ATUS_dataset_edit)[ncol(ATUS_dataset_edit)] <- paste0("tu0", i)
  assign(paste0("part_tu0", i), 100 * (get(paste0("tu0",i)) > 0))
  ATUS_dataset_edit <- bind_cols(ATUS_dataset_edit, part_tu0 = get(paste0("part_tu0", i)))
  names(ATUS_dataset_edit)[ncol(ATUS_dataset_edit)] <- paste0("part_tu0", i)
}
for (i in c(10:16, 18)){
  assign(paste0("tu", i), ATUS_dataset_edit[,grep(paste0("^t", i), names(ATUS_dataset_edit))])
  assign(paste0("tu",i), rowSums(get(paste0("tu",i))))
  ATUS_dataset_edit <- bind_cols(ATUS_dataset_edit, tu = get(paste0("tu", i)))
  names(ATUS_dataset_edit)[ncol(ATUS_dataset_edit)] <- paste0("tu", i)
  assign(paste0("part_tu", i), 100 * (get(paste0("tu",i)) > 0))
  ATUS_dataset_edit <- bind_cols(ATUS_dataset_edit, part_tu0 = get(paste0("part_tu", i)))
  names(ATUS_dataset_edit)[ncol(ATUS_dataset_edit)] <- paste0("part_tu", i)
}

# Filter into two separate data sets, saving odd months (including July) for validation purposes
ATUS_training_data <- filter(ATUS_dataset_edit, month(TUDIARYDATE) %in% c(2,4,6,8,10,12))
ATUS_validation_data <- filter(ATUS_dataset_edit, month(TUDIARYDATE) %in% c(1,3,5,7,9,11))
ATUS_plotting_data <- filter(ATUS_dataset_edit, month(TUDIARYDATE) != 7)

# Remove columns with -1's, -3's, -4's. We do this instead of removing rows as we would only be l
#remove full time vs part time as has -1 (35304 times)
Data_minus_columns <- select(ATUS_training_data, -one_of("TRDPFTPT"))
#remove full time vs part time spouse as has -1 (59784 times)
Data_minus_columns <- select(Data_minus_columns, -one_of("TRSPFTPT"))
#remove age of youngest child as has -1 (51103 times)
Data_minus_columns <- select(Data_minus_columns, -one_of("TRYHHCHILD"))
#remove 2 jobs within last 7 days as has -1 (35304 times)
Data_minus_columns <- select(Data_minus_columns, -one_of("TEMJOT"))
#remove enrolled at high school etc as has -1 and -3 and also question about which (40739 and 846
Data_minus_columns <- select(Data_minus_columns, -one_of(c("TESCHENR", "TESCHLVL")))
#remove employment status of spouse as has -1 (43791 times)
Data_minus_columns <- select(Data_minus_columns, -one_of("TESPEMPNOT"))
```

```
#remove weekly earnings as has -1 (41936 times)
Data_minus_columns <- select(Data_minus_columns, -one_of("TRERNWA"))
#remove total hours usually worked per week due to -1 and -4 for vary
Data_minus_columns <- select(Data_minus_columns, -one_of("TEHRUSLT"))
```

The Compelling Change in Caring for & Helping Non-HH Members

Exploratory Data Analysis

The weighted means for each year are calculated for the 17 activity groups, both in minutes and % participation.

```
Data_minus_columns_year_grouped <- Data_minus_columns %>% group_by(TUYEAR)
weighted_means_data <- as_tibble(data.frame(TUYEAR = 2003:2017))
for (i in 1:9){
  weighted_means_data <- full_join(weighted_means_data, summarise_at(Data_minus_columns_year_grou
}
for (i in c(10:16, 18)){
  weighted_means_data <- full_join(weighted_means_data, summarise_at(Data_minus_columns_year_grou
}
part_weighted_means_data <- as_tibble(data.frame(TUYEAR = 2003:2017))
for (i in 1:9){
  part_weighted_means_data <- full_join(part_weighted_means_data, summarise_at(Data_minus_columns
}
for (i in c(10:16, 18)){
  part_weighted_means_data <- full_join(part_weighted_means_data, summarise_at(Data_minus_columns
}
```

Create and output some summary plots for the data in this state to aid in EDA.

```
melted_participation_data <- melt(select(part_weighted_means_data, "TUYEAR", grep("^part_tu", nam
melted_data <- melt(select(weighted_means_data, "TUYEAR", grep("^tu", names(weighted_means_data))
```

```
part_summary_plot <- ggplot(melted_participation_data, aes(x=TUYEAR, y=value, color = variable))
summary_plot <- ggplot(melted_data, aes(x=TUYEAR, y=value, color=variable)) + geom_smooth(se=FALS
```

```
print(part_summary_plot)
print(summary_plot)
```

The ATUS data set records time spent on activities in minutes. This report initially looks at how the proportion of American's partaking in the 17 different groups of activities has changed. The following table gives a summary of some of these changes. The activities included are those with a percentage change of over 10% and a variance greater than 0.5.

```
variance <- sapply(weighted_means_data, function(col) var(col))
percentage_change <- (weighted_means_data[15,]/weighted_means_data[1,] - 1)*100
Activity_year_measures <- bind_rows(variance, percentage_change)
Activity_year_measures <- bind_cols(as_tibble(data.frame(Measure = c("Variance", "% Change"))), s
part_variance <- sapply(part_weighted_means_data, function(col) var(col))
part_percentage_change <- (part_weighted_means_data[15,]/part_weighted_means_data[1,] - 1)*100
```

```
part_Activity_year_measures <- bind_rows(part_variance, part_percentage_change)
part_Activity_year_measures <- bind_cols(as_tibble(data.frame(Measure = c("Variance", "% Change")
```

```
kable_names <- "Measure"
for (i in 1:9){
  kable_names <- c(kable_names, paste0("tu0", i))
}
for (i in c(10:16, 18)){
  kable_names <- c(kable_names, paste0("tu", i))
}
names(part_Activity_year_measures) <- kable_names
part_Activity_year_measures_values <- select(part_Activity_year_measures, -"Measure") %>% mutate(
part_Activity_year_measures_values <- part_Activity_year_measures_values %>% mutate(row_n = 1:n()
part_Activity_year_measures <- bind_cols(select(part_Activity_year_measures, "Measure"), part_Act
part_tu03_kable <- kable(part_Activity_year_measures, caption = "Change in Participation of Activ
kable_styling(part_tu03_kable, full_width = T, latex_options = c("hold_position"))
```

Table 1: Change in Participation of Activities

Measure	tu04	tu08	tu13	tu14	tu16
Variance	2.54	0.89	1.44	0.60	1.93
% Change	-32.11	-26.46	10.17	12.32	-24.08

Best subsets regression on a linear model is performed to estimate which variables have the largest effect on `tu04` participation percentage.

A summary spline curve glm with log link and multiplicative errors is fitted to show the change in `tu04` over the period.

```
n1 <- nrow(part_weighted_means_data)
part_tu04_summary_model <- glm(part_tu04 ~ ns(TUYEAR, knots = seq(2004, 2016, 2)), data = part_we
                           family = quasi(link = "log", variance = "mu^2"),
                           mustart = rep(5, n1))
part_tu04_summary_model_data <- as_tibble(data.frame(TUYEAR=seq(2003, 2017, length.out = 1000)))
part_tu04_summary_predicted <- predict(part_tu04_summary_model,
                                       newdata = part_tu04_summary_model_data,
                                       type = "link", se = TRUE)
part_tu04_summary_predicted <-bind_cols(part_tu04_summary_model_data,
                                as_tibble(data.frame(fitted = exp(part_tu04_summary_predicted$fit)
                                                     ymin = exp(part_tu04_summary_predicted$fit-1.
                                                     ymax = exp(part_tu04_summary_predicted$fit+1.
part_tu04_summary_plot <- ggplot(part_weighted_means_data, aes(x=TUYEAR, y=part_tu04)) +
  geom_line(data = part_tu04_summary_predicted, aes(y=fitted), size=1) +
  geom_ribbon(data = part_tu04_summary_predicted, aes(y=fitted, ymin=ymin, ymax=ymax), alpha=0.1)
```

```
print(part_tu04_summary_plot)
```

Best subsets showed that sex and number of household children have the largest effect on `tu04` participation %. Therefore `tu04` participation % is first broken down by sex; then new weighted means are calculated.

```
Data_minus_columns_year_sex_grouped <- Data_minus_columns %>% group_by(TUYEAR, TESEX)
sex_weighted_means_data <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.factor(c("M", "
for (i in 1:9){
  sex_weighted_means_data <- full_join(sex_weighted_means_data, summarise_at(Data_minus_columns_y
}
for (i in c(10:16, 18)){
  sex_weighted_means_data <- full_join(sex_weighted_means_data, summarise_at(Data_minus_columns_y
}
part_sex_weighted_means_data <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.factor(c("
for (i in 1:9){
  part_sex_weighted_means_data <- full_join(part_sex_weighted_means_data, summarise_at(Data_minus
}
for (i in c(10:16, 18)){
  part_sex_weighted_means_data <- full_join(part_sex_weighted_means_data, summarise_at(Data_minus
}
```

A new model is built to take this split into account. The model is built using familiar techniques from previous lab reports.

```
n2 <- nrow(part_sex_weighted_means_data)
part_tu04_sex_model <- glm(part_tu04 ~ -1 + TESEX + TESEX:ns(TUYEAR, knots = seq(2004, 2016, 2)),
                           family = quasi(link = "log", variance = "mu^2"),
                           mustart = rep(5, n2))
part_tu04_sex_model_data <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017, length.out = 1000), TESE
part_tu04_sex_predicted <- predict(part_tu04_sex_model, newdata = part_tu04_sex_model_data, type
part_tu04_sex_predicted <-bind_cols(part_tu04_sex_model_data,
                                    as_tibble(data.frame(fitted = exp(part_tu04_sex_predicted$fit),
                                                         ymin = exp(part_tu04_sex_predicted$fit-1.128*
                                                         ymax = exp(part_tu04_sex_predicted$fit+1.128*
part_tu04_sex_plot <- ggplot(part_sex_weighted_means_data, aes(x=TUYEAR, y=part_tu04)) +
  geom_line(data = part_tu04_sex_predicted, aes(y=fitted, col=TESEX), size=1) +
  geom_ribbon(data = filter(part_tu04_sex_predicted, TESEX =="M"), aes(y=fitted, ymin=ymin, ymax=
  geom_ribbon(data = filter(part_tu04_sex_predicted, TESEX =="F"), aes(y=fitted, ymin=ymin, ymax=
```

```
print(part_tu04_sex_plot)
```

Weighted means for the number of household children for each group of people and year are calculated in order to see the change over the period. A bar chart of these weighted mean number of household children is added to plot to provide visual aid in observing the trend.

```
sex_weighted_means_TRCHILDNUM <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.factor(c(
for (i in 1:9){
  sex_weighted_means_TRCHILDNUM <- full_join(sex_weighted_means_TRCHILDNUM, summarise_at(Data_min
}
for (i in c(10:16, 18)){
  sex_weighted_means_TRCHILDNUM <- full_join(sex_weighted_means_TRCHILDNUM, summarise_at(Data_min
}
part_sex_weighted_means_TRCHILDNUM <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.fact
for (i in 1:9){
  part_sex_weighted_means_TRCHILDNUM <- full_join(part_sex_weighted_means_TRCHILDNUM, summarise_a
}
```

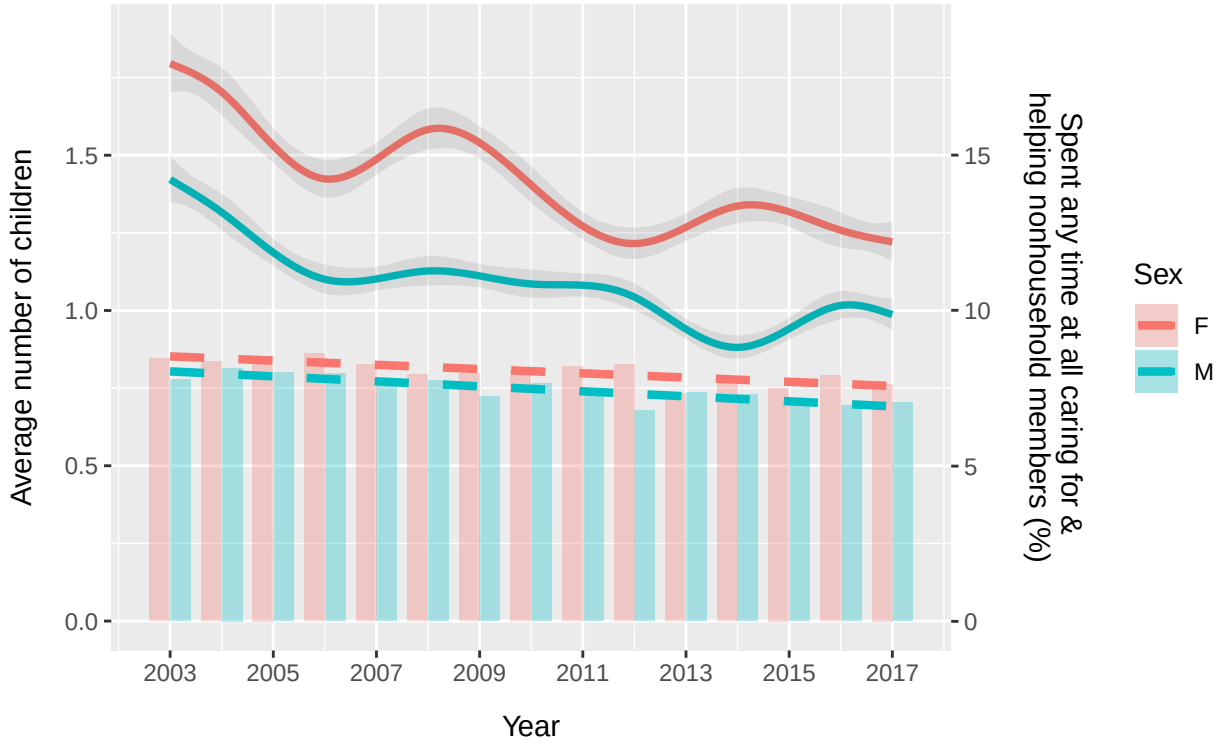
```
for (i in c(10:16, 18)){
  part_sex_weighted_means_TRCHILDNUM <- full_join(part_sex_weighted_means_TRCHILDNUM, summarise_a
}
part_tu04_sex_TRCHILDNUM_plot <- ggplot(part_sex_weighted_means_data, aes(x=TUYEAR, y=part_tu04))
  geom_bar(data=part_sex_weighted_means_TRCHILDNUM, aes(x=as.numeric(as.character(TUYEAR)), y=TRC
  geom_smooth(method = "lm", data=part_sex_weighted_means_TRCHILDNUM, aes(x=as.numeric(as.charact
  geom_line(data = part_tu04_sex_predicted, aes(x=TUYEAR, y=0.1*fitted, col=TESEX), size=1.3) +
  geom_ribbon(data = filter(part_tu04_sex_predicted, TESEX =="M"), aes(y=0.1*fitted, ymin=0.1*ymi
  geom_ribbon(data = filter(part_tu04_sex_predicted, TESEX =="F"), aes(y=0.1*fitted, ymin=0.1*ymi
  scale_y_continuous(sec.axis = sec_axis(~.*(1/0.1), name = "Spent any time at all caring for & \
  scale_x_continuous(breaks = seq(2003,2017,2)) + xlab("Year") + ylab("Average number of children
```

The suitability of the model is checked using an F-test. It shows there is a significant improvement over the original model.

```
part_tu04_summary_model_anova <- glm(part_tu04 ~ ns(TUYEAR, knots = seq(2004, 2016, 2)), data = p
                                     mustart = rep(5, n2))
ANOVA_for_sex <- anova(part_tu04_summary_model_anova, part_tu04_sex_model, test="F")
```

The final plot is printed.

```
print(part_tu04_sex_TRCHILDNUM_plot)
```



The conclusion from the EDA in regards to the participation % for `tu04` is informally tested by creating a simpler linear model which when tested shows that the coefficient for Year is < 0 and therefore confirms the belief.

```
part_tu04_sex_linear_model <- lm(data=part_sex_weighted_means_data, part_tu04~TUYEAR + TESEX)
part_tu04_sex_linear_model_summary <- summary(part_tu04_sex_linear_model)
Year_t <- coefficients(part_tu04_sex_linear_model_summary)[2,3]
lm_p_value <- pt(Year_t, part_tu04_sex_linear_model$df, lower=TRUE)
```

In order to check there is a link between the number of household children a new model must be produced which includes this as an explanatory variable. The code below builds this model.

```
#note we have only 1439 respondents with TRCHILDNUM>4 so we set these to 4 so that the algorithm
Data_minus_columns_year_sex_child_grouped <- Data_minus_columns
Data_minus_columns_year_sex_child_grouped <- mutate(Data_minus_columns_year_sex_child_grouped, TR
sex_child_weighted_means_data <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.factor(c(
for (i in 1:9){
  sex_child_weighted_means_data <- full_join(sex_child_weighted_means_data, summarise_at(Data_min
}
for (i in c(10:16, 18)){
  sex_child_weighted_means_data <- full_join(sex_child_weighted_means_data, summarise_at(Data_min
}
part_sex_child_weighted_means_data <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.fact
for (i in 1:9){
  part_sex_child_weighted_means_data <- full_join(part_sex_child_weighted_means_data, summarise_a
}
for (i in c(10:16, 18)){
  part_sex_child_weighted_means_data <- full_join(part_sex_child_weighted_means_data, summarise_a
}
n3 <- nrow(part_sex_child_weighted_means_data)
part_tu04_sex_child_model <- glm(part_tu04 ~ -1 + TRCHILDNUM + TESEX + TESEX:ns(TUYEAR, knots = s
                          family = quasi(link = "log", variance = "mu^2"),
                          mustart = rep(5, n3))
```

A t-test confirms that this variable is significant, implying there is a dependence.

```
sex_child_summary_for_t_test <- summary(part_tu04_sex_child_model)
sex_child_p_value <- coefficients(sex_child_summary_for_t_test)[1,4]
```

The key observation is that, as the table suggested, over the period the participation in caring for & helping non-household members has decreased for both men and women. Building a linear model using `TESEX` and `TUYEAR` and performing a t-test confirms this. There also appears to be a link between participation in caring for & helping non-household members and the number of household children. Updating the spline curve model to include the number of household children and then again testing this hypothesis with a t-test confirms this is the case. Looking at the graph, fluctuations in the average number of household children, on the whole seem to be followed by but if we check the correlations between them, they show this link is fairly weak (0.65 for Men and 0.49 for Women), suggesting there must be further reasons for this change; possibly not measured in this dataset.

Validation

In order to make formal conclusions based on the EDA we must define hypotheses and test them on the validation data.

The same tidying procedure is completed on the validation data.

```
#remove full time vs part time as has -1
Data_minus_columns_validation <- select(ATUS_validation_data, -one_of("TRDPFTPT"))
#remove full time vs part time spouse as has -1
```

```
Data_minus_columns_validation <- select(Data_minus_columns_validation, -one_of("TRSPFTPT"))
#remove age of youngest child as has -1
Data_minus_columns_validation <- select(Data_minus_columns_validation, -one_of("TRYHHCHILD"))
#remove 2 jobs within last 7 days as has -1
Data_minus_columns_validation <- select(Data_minus_columns_validation, -one_of("TEMJOT"))
#remove enrolled at high school etc as has -1 and -3 and also question about which
Data_minus_columns_validation <- select(Data_minus_columns_validation, -one_of(c("TESCHENR", "TES
#remove employment status of spouse as has -1
Data_minus_columns_validation <- select(Data_minus_columns_validation, -one_of("TESPEMPNOT"))
#remove weekly earnings as has -1
Data_minus_columns_validation <- select(Data_minus_columns_validation, -one_of("TRERNWA"))
#remove total hours usually worked per week due to -1 and -4 for vary
Data_minus_columns_validation <- select(Data_minus_columns_validation, -one_of("TEHRUSLT"))
```

Again, weighted means are calculated for men and women; for each year.

```
Data_minus_columns_year_sex_grouped_validation <- Data_minus_columns_validation %>% group_by(TUYE
sex_weighted_means_data_validation <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.fact
for (i in 1:9){
  sex_weighted_means_data_validation <- full_join(sex_weighted_means_data_validation, summarise_a
}
for (i in c(10:16, 18)){
  sex_weighted_means_data_validation <- full_join(sex_weighted_means_data_validation, summarise_a
}
part_sex_weighted_means_data_validation <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as
for (i in 1:9){
  part_sex_weighted_means_data_validation <- full_join(part_sex_weighted_means_data_validation, s
}
for (i in c(10:16, 18)){
  part_sex_weighted_means_data_validation <- full_join(part_sex_weighted_means_data_validation, s
}
```

A linear model is fitted to the validation data and now a formal t-test is completed. It confirms the proportion of American's who participate in caring for & helping non-household members has decreased over the period.

```
part_tu04_sex_linear_model_validation <- lm(data=part_sex_weighted_means_data_validation, part_tu
part_tu04_sex_linear_model_summary_validation <- summary(part_tu04_sex_linear_model_validation)
Year_t_validation <- coefficients(part_tu04_sex_linear_model_summary_validation)[2,3]
lm_p_value_validation <- pt(Year_t_validation, part_tu04_sex_linear_model_validation$df, lower=TR
```

Like with the EDA, a spline curve model is fitted.

```
n2_validation <- nrow(part_sex_weighted_means_data_validation)
part_tu04_sex_model_validation <- glm(part_tu04 ~ -1 + TESEX + TESEX:ns(TUYEAR, knots = seq(2004,
                          family = quasi(link = "log", variance = "mu^2"),
                          mustart = rep(5, n2_validation))
part_tu04_sex_model_data_validation <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017, length.out =
part_tu04_sex_predicted_validation <- predict(part_tu04_sex_model_validation, newdata = part_tu04
part_tu04_sex_predicted_validation <-bind_cols(part_tu04_sex_model_data_validation,
                                               as_tibble(data.frame(fitted = exp(part_tu04_sex_pr
                                                                    ymin = exp(part_tu04_sex_pred
```

```
                                                                    ymax = exp(part_tu04_sex_pred
part_tu04_sex_plot_validation <- ggplot(part_sex_weighted_means_data_validation, aes(x=TUYEAR, y=
  geom_line(data = part_tu04_sex_predicted_validation, aes(y=fitted, col=TESEX), size=1) +
  geom_ribbon(data = filter(part_tu04_sex_predicted_validation, TESEX =="M"), aes(y=fitted, ymin=
  geom_ribbon(data = filter(part_tu04_sex_predicted_validation, TESEX =="F"), aes(y=fitted, ymin=
```

Again, to check if the link between number of household children and tu04 participation % is significant it must be tested using an update of the model which includes the number of household children as an explanatory variable.

```
Data_minus_columns_year_sex_child_grouped_validation <- Data_minus_columns_validation
Data_minus_columns_year_sex_child_grouped_validation <- mutate(Data_minus_columns_year_sex_child_
sex_child_weighted_means_data_validation <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=a
for (i in 1:9){
  sex_child_weighted_means_data_validation <- full_join(sex_child_weighted_means_data_validation,
}
for (i in c(10:16, 18)){
  sex_child_weighted_means_data_validation <- full_join(sex_child_weighted_means_data_validation,
}
part_sex_child_weighted_means_data_validation <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TE
for (i in 1:9){
  part_sex_child_weighted_means_data_validation <- full_join(part_sex_child_weighted_means_data_v
}
for (i in c(10:16, 18)){
  part_sex_child_weighted_means_data_validation <- full_join(part_sex_child_weighted_means_data_v
}
n3_validation <- nrow(part_sex_child_weighted_means_data_validation)
part_tu04_sex_child_model_validation <- glm(part_tu04 ~ -1 + TRCHILDNUM + TESEX + TESEX:ns(TUYEAR
                          family = quasi(link = "log", variance = "mu^2"),
                          mustart = rep(5, n3_validation))
```

The t-test is completed on the model and confirms the belief of the presence of a link.

```
sex_child_summary_for_t_test_validation <- summary(part_tu04_sex_child_model_validation)
sex_child_p_value_validation <- coefficients(sex_child_summary_for_t_test_validation)[1,4]
```

To plot the results, all of the data, excluding those for July, is used. The same method of tidying the data, calculating weighted means, building a spline curve glm model with log link and multiplicative error and plotting a graphical interpretation of the model is performed.

```
#remove full time vs part time as has -1
Data_minus_columns_plotting <- select(ATUS_plotting_data, -one_of("TRDPFTPT"))
#remove full time vs part time spouse as has -1
Data_minus_columns_plotting <- select(Data_minus_columns_plotting, -one_of("TRSPFTPT"))
#remove age of youngest child as has -1
Data_minus_columns_plotting <- select(Data_minus_columns_plotting, -one_of("TRYHHCHILD"))
#remove 2 jobs within last 7 days as has -1
Data_minus_columns_plotting <- select(Data_minus_columns_plotting, -one_of("TEMJOT"))
#remove enrolled at high school etc as has -1 and -3 and also question about which
Data_minus_columns_plotting <- select(Data_minus_columns_plotting, -one_of(c("TESCHENR", "TESCHLV
#remove employment status of spouse as has -1
Data_minus_columns_plotting <- select(Data_minus_columns_plotting, -one_of("TESPEMPNOT"))
```

```r
#remove weekly earnings as has -1
Data_minus_columns_plotting <- select(Data_minus_columns_plotting, -one_of("TRERNWA"))
#remove total hours usually worked per week due to -1 and -4 for vary
Data_minus_columns_plotting <- select(Data_minus_columns_plotting, -one_of("TEHRUSLT"))
```

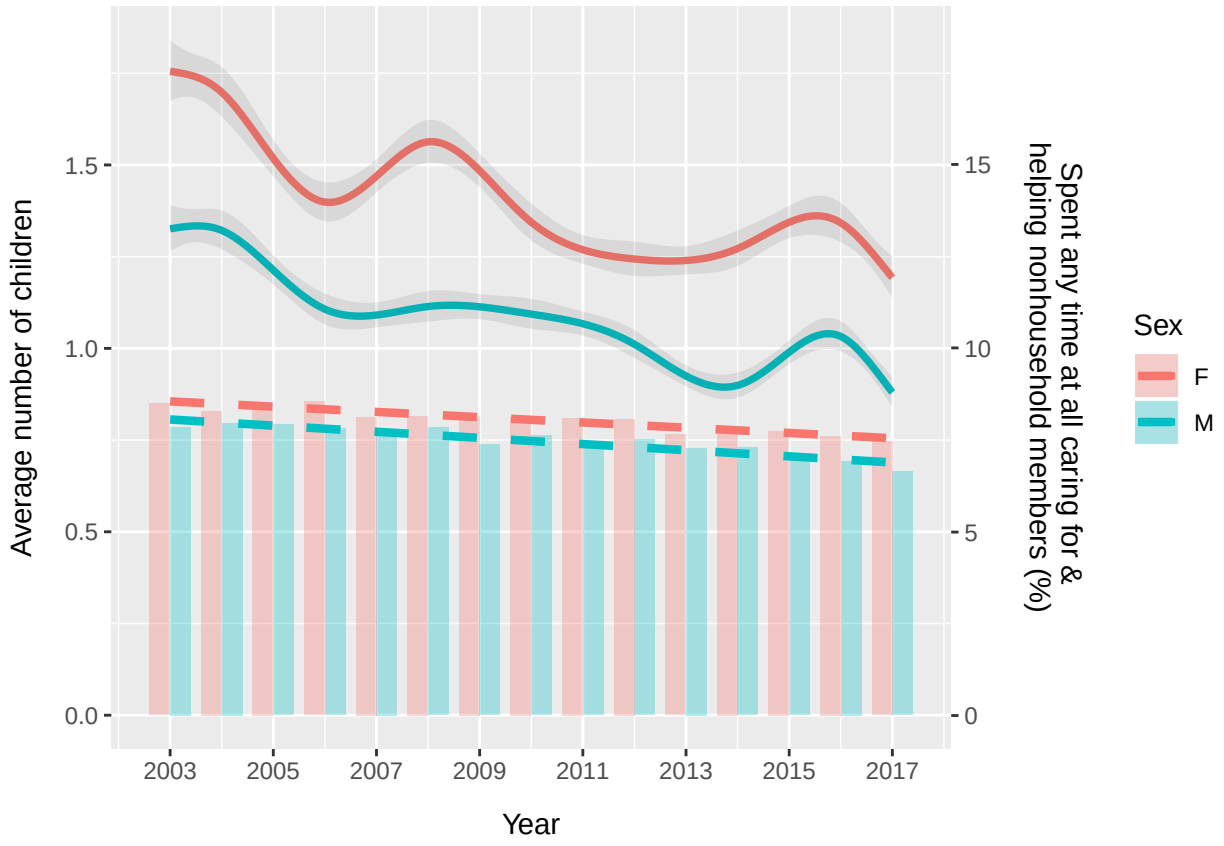
```r
Data_minus_columns_year_sex_grouped_plotting <- Data_minus_columns_plotting %>% group_by(TUYEAR,
sex_weighted_means_data_plotting <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.factor
for (i in 1:9){
  sex_weighted_means_data_plotting <- full_join(sex_weighted_means_data_plotting, summarise_at(Da
}
for (i in c(10:16, 18)){
  sex_weighted_means_data_plotting <- full_join(sex_weighted_means_data_plotting, summarise_at(Da
}
part_sex_weighted_means_data_plotting <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.f
for (i in 1:9){
  part_sex_weighted_means_data_plotting <- full_join(part_sex_weighted_means_data_plotting, summa
}
for (i in c(10:16, 18)){
  part_sex_weighted_means_data_plotting <- full_join(part_sex_weighted_means_data_plotting, summa
}
```

```r
n2_plotting <- nrow(part_sex_weighted_means_data_plotting)
part_tu04_sex_model_plotting <- glm(part_tu04 ~ -1 + TESEX + TESEX:ns(TUYEAR, knots = seq(2004, 2
                          family = quasi(link = "log", variance = "mu^2"),
                          mustart = rep(5, n2_plotting))
part_tu04_sex_model_data_plotting <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017, length.out = 10
part_tu04_sex_predicted_plotting <- predict(part_tu04_sex_model_plotting, newdata = part_tu04_sex
part_tu04_sex_predicted_plotting <-bind_cols(part_tu04_sex_model_data_plotting,
                                             as_tibble(data.frame(fitted = exp(part_tu04_sex_pr
                                                                  ymin = exp(part_tu04_sex_pred
                                                                  ymax = exp(part_tu04_sex_pred
```

```r
sex_weighted_means_TRCHILDNUM_plotting <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESEX=as.
for (i in 1:9){
  sex_weighted_means_TRCHILDNUM_plotting <- full_join(sex_weighted_means_TRCHILDNUM_plotting, sum
}
for (i in c(10:16, 18)){
  sex_weighted_means_TRCHILDNUM_plotting <- full_join(sex_weighted_means_TRCHILDNUM_plotting, sum
}
part_sex_weighted_means_TRCHILDNUM_plotting <- as_tibble(expand.grid(TUYEAR=seq(2003, 2017), TESE
for (i in 1:9){
  part_sex_weighted_means_TRCHILDNUM_plotting <- full_join(part_sex_weighted_means_TRCHILDNUM_plo
}
for (i in c(10:16, 18)){
  part_sex_weighted_means_TRCHILDNUM_plotting <- full_join(part_sex_weighted_means_TRCHILDNUM_plo
}
part_tu04_sex_TRCHILDNUM_plot_plotting <- ggplot(part_sex_weighted_means_data_plotting, aes(x=TUY
  geom_bar(data=part_sex_weighted_means_TRCHILDNUM_plotting, aes(x=as.numeric(as.character(TUYEAR
  geom_smooth(method = "lm", data=part_sex_weighted_means_TRCHILDNUM_plotting, aes(x=as.numeric(a
  geom_line(data = part_tu04_sex_predicted_plotting, aes(x=TUYEAR, y=0.1*fitted, col=TESEX), size
```

```
  geom_ribbon(data = filter(part_tu04_sex_predicted_plotting, TESEX =="M"), aes(y=0.1*fitted, ymi
  geom_ribbon(data = filter(part_tu04_sex_predicted_plotting, TESEX =="F"), aes(y=0.1*fitted, ymi
  scale_y_continuous(sec.axis = sec_axis(~.*(1/0.1), name = "Spent any time at all caring for & \
  scale_x_continuous(breaks = seq(2003,2017,2)) + xlab("Year") + ylab("Average number of children
```

```
print(part_tu04_sex_TRCHILDNUM_plot_plotting)
```



Investigating Whether the Time Spent on Traditionally Gendered Activities has Converged as Gender Roles have Broken Down

Preparing the Data and EDA

Apply some reformatting to the data imported previously to better fit the exploratory analysis to be carried out. This includes the creation of a `TUBIRTHYEAR` feature to be used in separating people into different generations (Millenials, Baby Boomers etc.) represented by another feature called `TUGENERATION`. The `ATUS.CPS` dataset is then filtered to select regional data which can be joined to the dataset created so far for EDA.

```
# Get birth year as a variable for purpose of classifying generation by subtratcting age from the
ATUS_dataset$TUBIRTHYEAR <- ATUS_dataset$TUYEAR - ATUS_dataset$TEAGE
ATUS_dataset$TUGENERATION <- ifelse(ATUS_dataset$TUBIRTHYEAR < 1946, "Silent Generation",
    ifelse(ATUS_dataset$TUBIRTHYEAR < 1965, "Baby Boomers",
        ifelse(ATUS_dataset$TUBIRTHYEAR < 1981, "Generation X", "Millennials")))
# Filter CPS down to just relevant variables. GEDSTFIPS is state, GEDIV is division and GEREG is
```

```
ATUS.CPS.States <- select(ATUS.CPS, TUCASEID, GESTFIPS, GEDIV, GEREG)
ATUS.CPS.States <- distinct(ATUS.CPS.States)
```

Next, collections of variable representing traditional gender roles can be extracted. `ATUS_dataset` can then be subsetted to 'Gender.Roles.Data', including only relevant variables. Each of the collections of gender role variables can be appended to this dataset by summing over all of the relevant sub-variables. E.g. `t05` and all suffixes of this expression represent time spent working, so these can be summed over and collected under one variable in `Gender.Roles.Data$Working`.

```
# Create a data frame from relevant variables within ATUS Summary
Gender.Roles.Data <- select(ATUS_dataset, TUCASEID, TESEX, TEAGE, TUFNWGTP, TUYEAR, TUDIARYDATE,
# Create variables to be added to a data frame with relevant gender activities
Working <- ATUS_dataset[, grep("^t05", names(ATUS_dataset))]
House.Maintenance <- ATUS_dataset[, grep("^t0204", names(ATUS_dataset))]
Vehicle.Maintenance <- ATUS_dataset[, grep("^t0207", names(ATUS_dataset))]
Housework <- ATUS_dataset[, grep("^t0201", names(ATUS_dataset))]
Food.Preparation <- ATUS_dataset[, grep("^t0202", names(ATUS_dataset))] + ATUS_dataset$t070101 +
Childcare <- ATUS_dataset[, grep("^t03", names(ATUS_dataset))]
# Add in relevant categories
Gender.Roles.Data$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.Preparation <- rowSums(Food.Preparation)
Gender.Roles.Data$Childcare <- rowSums(select(Childcare, t030101:t030399))
# Join this dataset with the CPS states info by unique id number of particiapnt
Gender.Roles.Data <- inner_join(Gender.Roles.Data, ATUS.CPS.States, by = 'TUCASEID')
```

This for-loop structure calculates weighted means for each possible combination of `variable`, `month`, `generation`, `region`, `year` and `sex`, which are to be used in modelling the data. The means are calculated for a demographic represented by one of these combinations and then attached to `Gender.Roles.Data` so that all of the individuals within that demographic / that conform to that combination have a weighted mean value for each variable.

This for loop also calculates a `Gender.Roles.Differences` data frame to store the differences between genders in the weighted mean values corresponding to each combination of `variable`, `month`, `generation` and `year`. These differences can then be used to perform statistical tests on the changes over the given period of data in the time use within each of the investigated variables to study how gender roles have changed.

```
for (variable in c("Working", "House.Maintenance", "Vehicle.Maintenance", "Housework", "Food.Prep
    Gender.Roles.Data[,paste0('weighted.', variable)] <- 0
    for (generation in c("Silent Generation", "Baby Boomers", "Generation X", "Millennials")) {
        for (year in unique(Gender.Roles.Data$TUYEAR)) {
            for (month in unique(month(Gender.Roles.Data$TUDIARYDATE))) {
                for (sex in c("M", "F")) {
                    for (region in 1:4) {
                        # Work only with the data which conforms to the current step in the for l
                        Filtered.Data <- filter(Gender.Roles.Data, TESEX == sex, TUYEAR == year,
                        bottom.sum <- sum(Filtered.Data$TUFNWGTP)
                        top.sum <- sum(Filtered.Data$TUFNWGTP * Filtered.Data[,variable])
```

```
                        # Use above calculations to work out weighted means for the current combi
                        weighted.values <- top.sum / bottom.sum
                        Gender.Roles.Data[Gender.Roles.Data$TESEX == sex & Gender.Roles.Data$TUYE
                    }
                }
            }
        }
    }
}
```

Split the full dataset into `Train`, `Validate` and `Plotting` subsets, where `Train` includes all of the even-numbered months, `Validate` contains all of the odd-numbered months. `Plotting` includes all of the months except July as required for the task, it is this data that is used for plots in the final report.

```
Gender.Roles.Train <- filter(Gender.Roles.Data, month(TUDIARYDATE) %% 2 == 0)
Gender.Roles.Validate <- filter(Gender.Roles.Data, month(TUDIARYDATE) %% 2 != 0)
Gender.Roles.Plotting <- filter(Gender.Roles.Data, month(TUDIARYDATE) != 7)
```

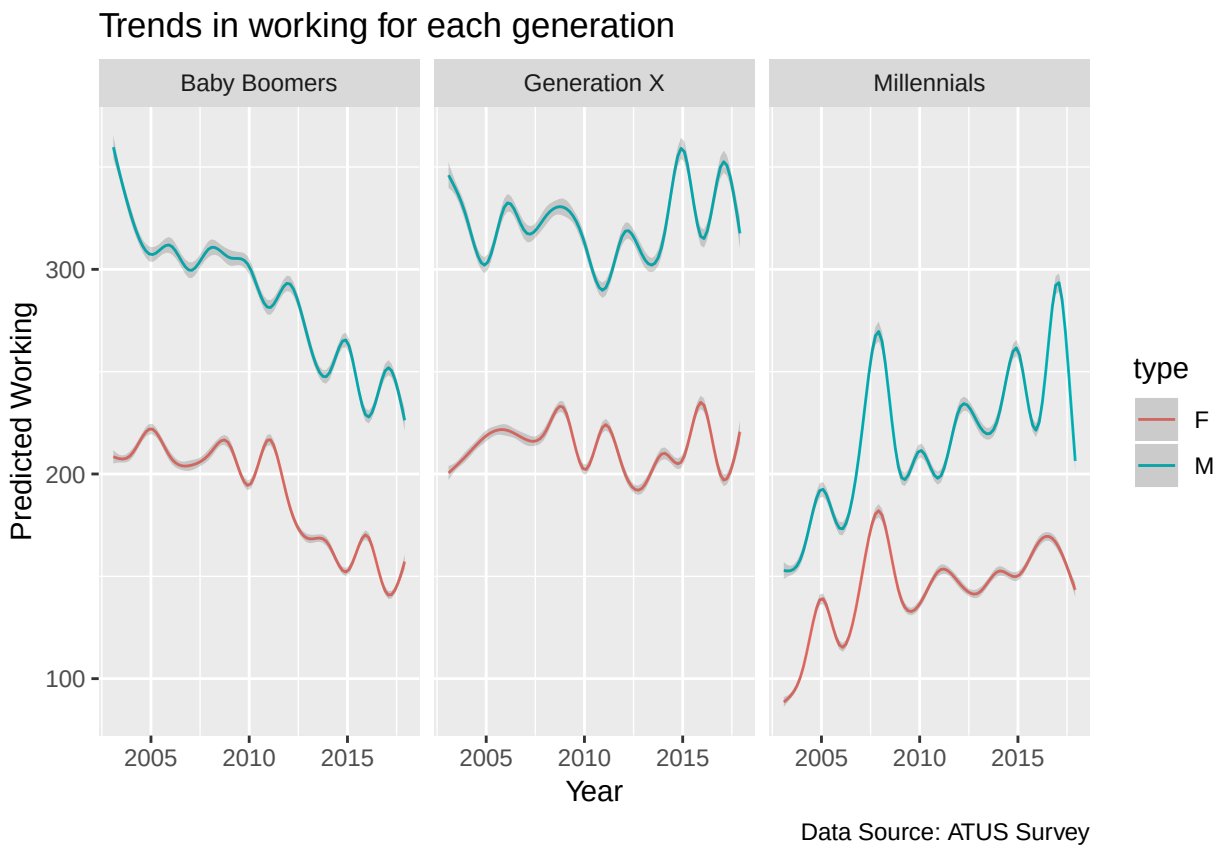
Below is a function which fits models on the passed dataset (one of the ones defined above) and then generates predictions with which plots can be made. This function is called multiple times below with appropriate `mu` start values passed to ensure the `glm()` algorithm converges. These plots are used extensively in the Gender Roles section of the report.

```
# Make a vector containing each year as a Date object to be used in generating splines for the mo
Dates = c()
for (i in 4:17) {
    if (i < 10) {
        Dates[i-3] <- paste(paste0("200", i), "01", "01", sep = "-")
    } else {
        Dates[i-3] <- paste(paste0("20", i), "01", "01", sep = "-")
    }
}

# Function to create glm()'s
fit.models.and.plot <- function(Data, variable, mu) {
    n <- nrow(Data)
    # Commented out models show alternative number of splines / setup
    # model1 <- glm(as.formula(paste0('weighted.',variable,' ~ ns(TUDIARYDATE, knots = (c(as.Date
    model1 <- glm(as.formula(paste0('weighted.',variable,' ~ ns(TUDIARYDATE)')), data = Data, fam

    # Similar model structure to that found in lab 5, works well in this context
    model2 <- update(model1, . ~ -1 + TESEX + TESEX:ns(TUDIARYDATE, knots = c(as.Date(Dates))))

    # UNUSED model was tested to investigate regional differences, it was decided that generation
    model3 <- update(model2, .~. + GEREG + GEREG:ns(TUDIARYDATE, knots = c(as.Date(Dates))))

    # Incorporating generational differences into the model allows for grouping and plots to inve
    model4 <- update(model2, .~. + TUGENERATION + TUGENERATION:ns(TUDIARYDATE, knots = c(as.Date(
    # model4 <- update(model2, .~. + GEREG + GEREG:ns(TUDIARYDATE, knots = c(as.Date(Dates))))
    # Use same method as in lab 5 to plot them
```

```
    # change the model number in the predict functions below for different models
    male.prediction <- predict.glm(model4, newdata = filter(Data, TESEX == 'M'), se = TRUE, type
    female.prediction <- predict.glm(model4, newdata = filter(Data, TESEX == 'F'), se = TRUE, typ
    df <- data.frame(age = c(filter(Data, TESEX == 'M')$TEAGE, filter(Data, TESEX == 'F')$TEAGE),
    if (variable == 'Working' | variable == 'Childcare') {
        print(ggplot(filter(df, generation != "Silent Generation"), aes(x=date, y=prediction, col
        geom_line() +
        geom_ribbon(aes(ymin = prediction - prediction_error, ymax = prediction + prediction_erro
    } else {
        print(ggplot(df, aes(x=date, y=prediction, colour=type)) +
        geom_line() +
        geom_ribbon(aes(ymin = prediction - prediction_error, ymax = prediction + prediction_erro
    }
}
```

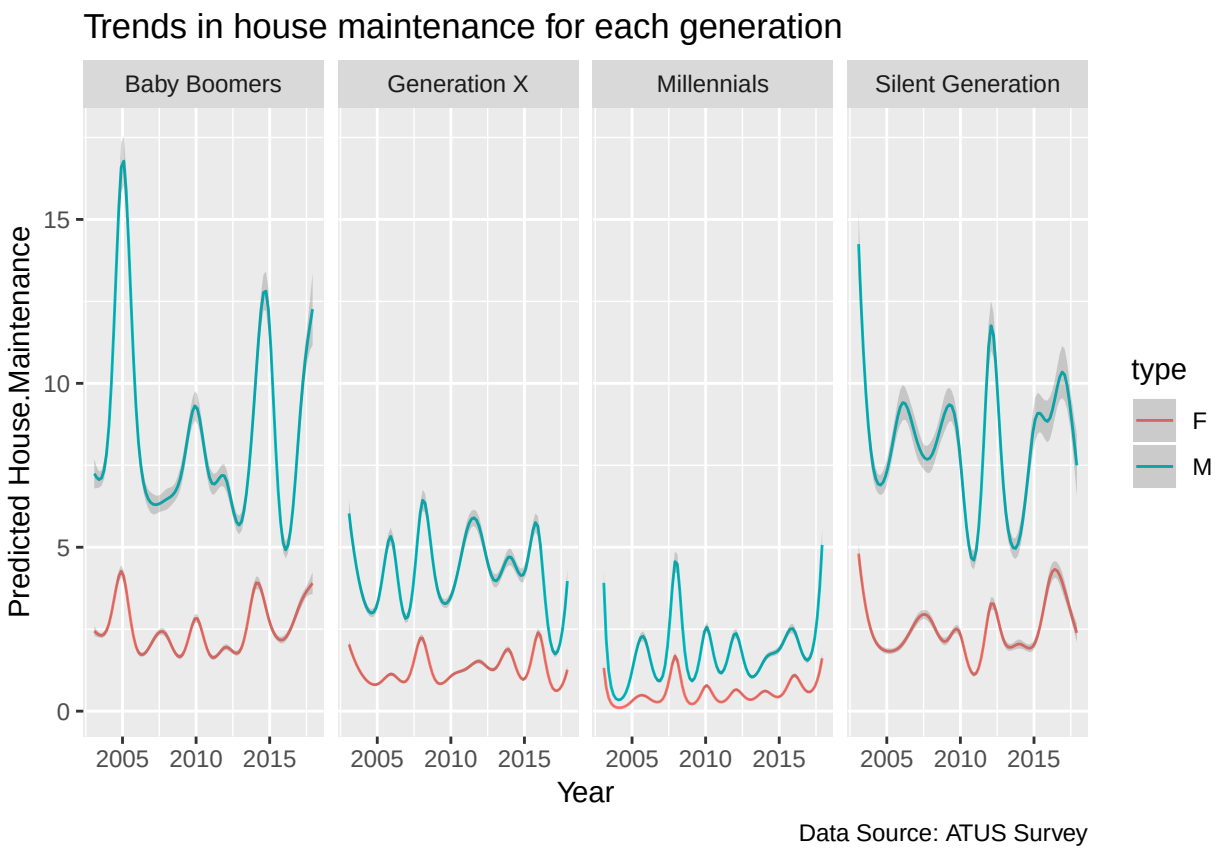
Generate plots for EDA to observe potential long-term trends over time for each of the chosen variables to investigate. `mu` values were experimented with once the data was first observed so that they roughly fall within the range of plotted values for each model. All of the plots show separate curves for male and female, as this was deemed to be more informative and interesting for the reader than plotting an engineered difference between the two.

```
fit.models.and.plot(Gender.Roles.Train, 'Working', 250)
```



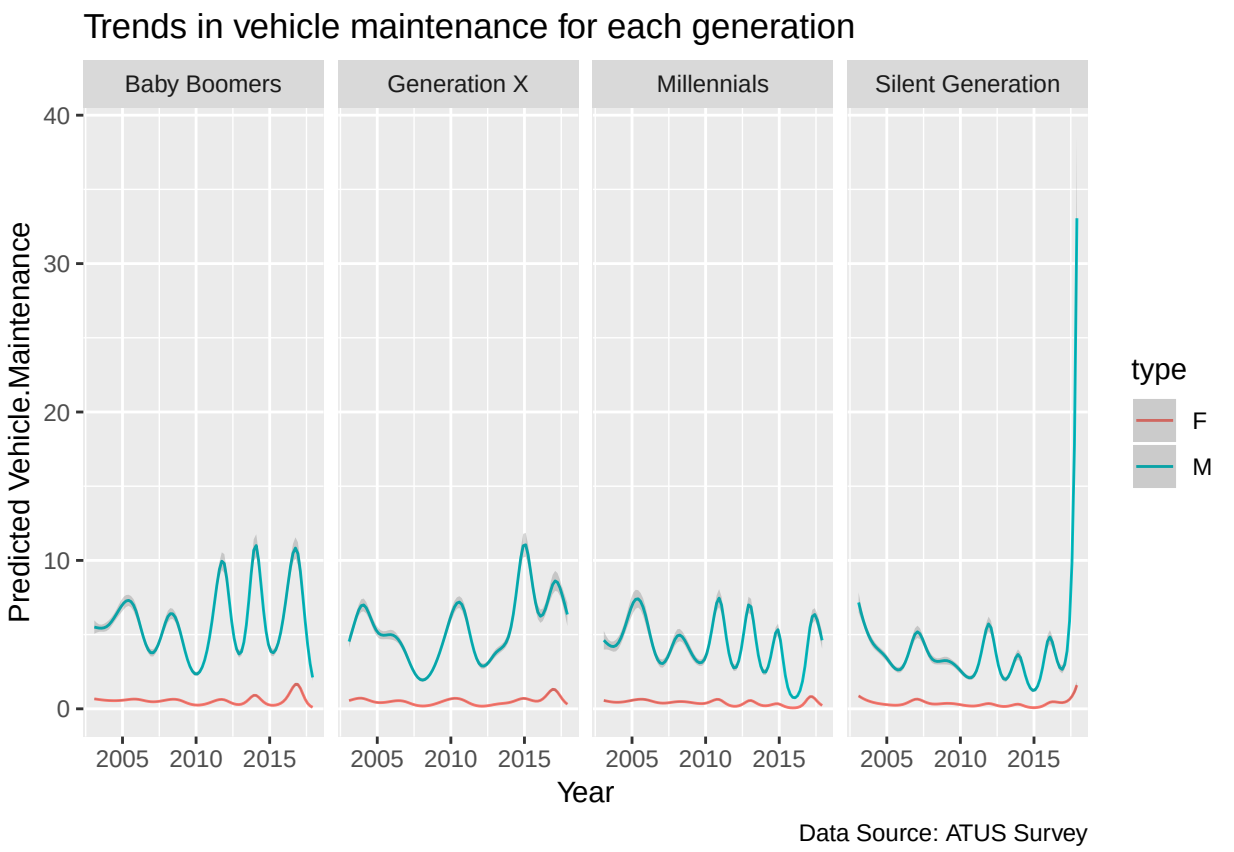

```
fit.models.and.plot(Gender.Roles.Train, 'House.Maintenance', 10)
```

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

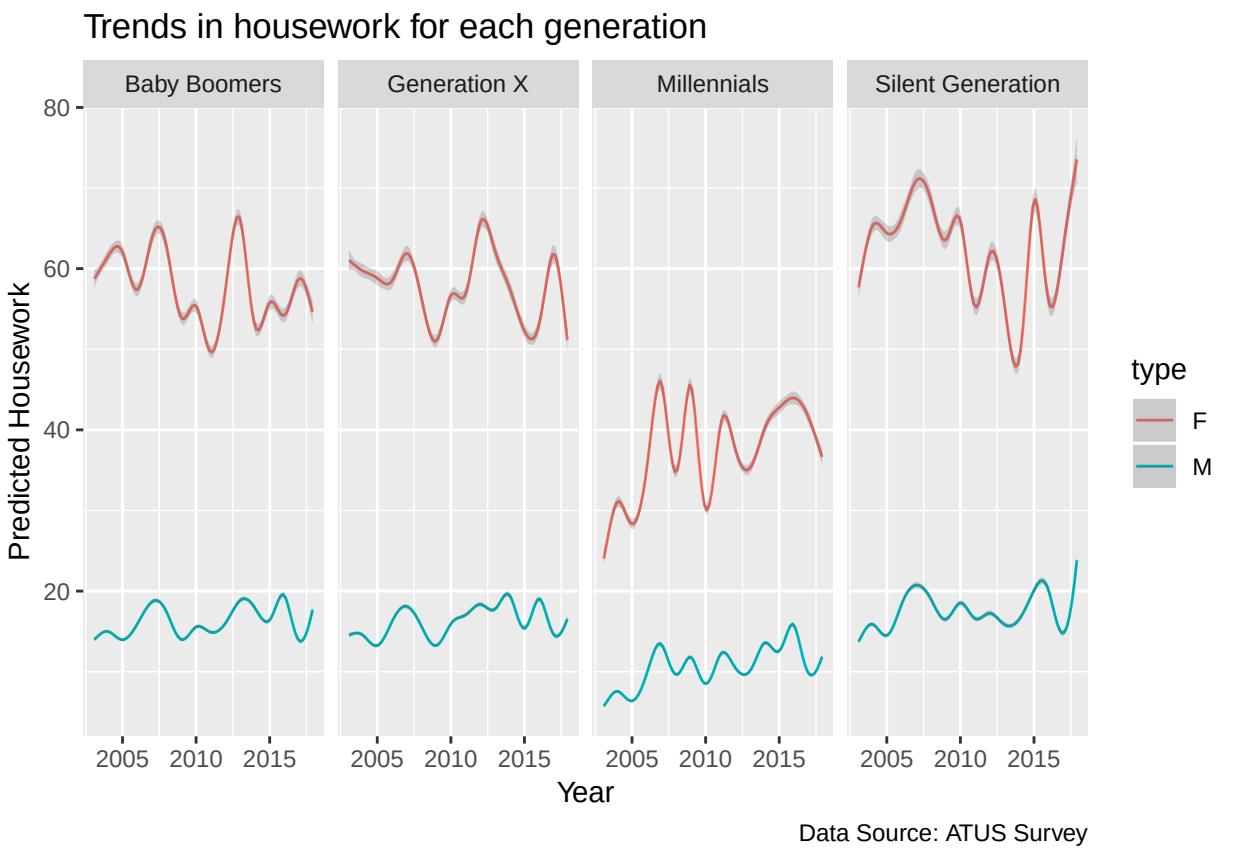



```
fit.models.and.plot(Gender.Roles.Train, 'Vehicle.Maintenance', 10)
```

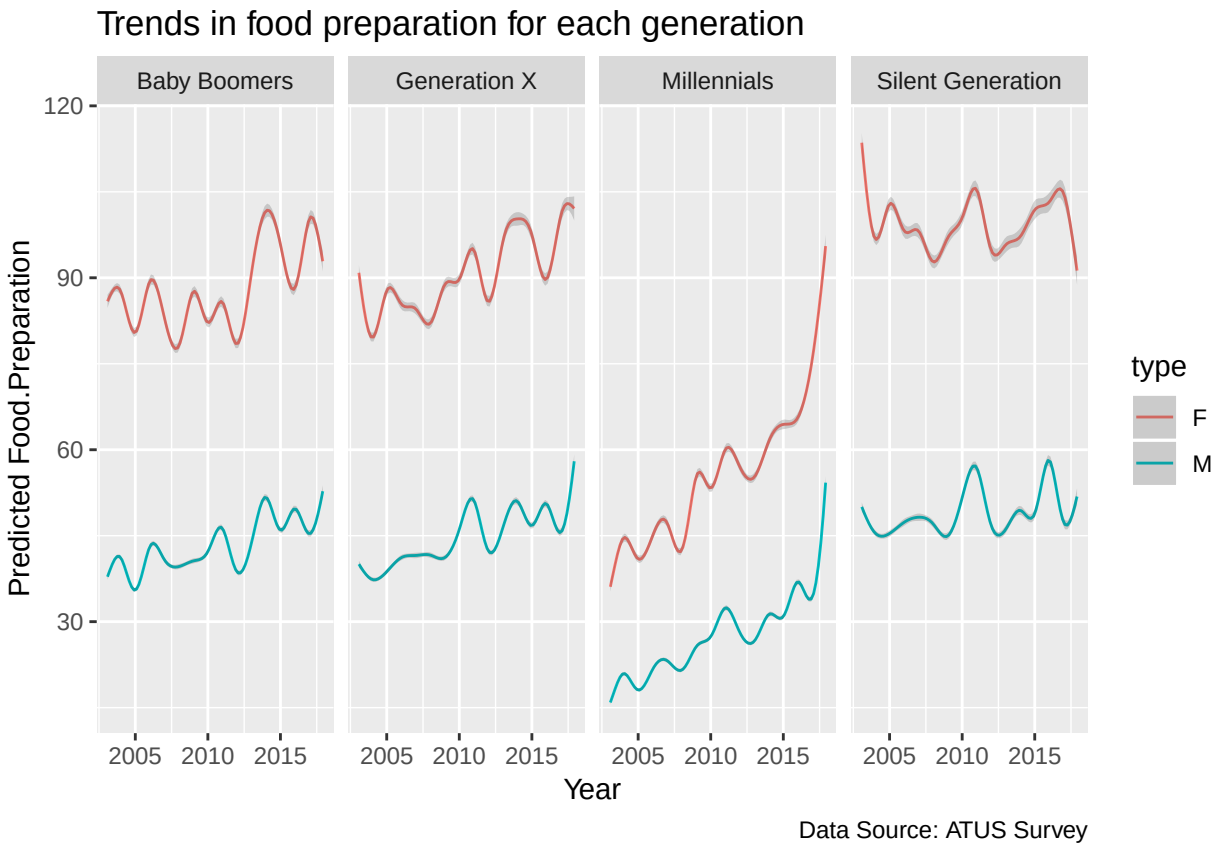
```
## Warning: glm.fit: algorithm did not converge
```



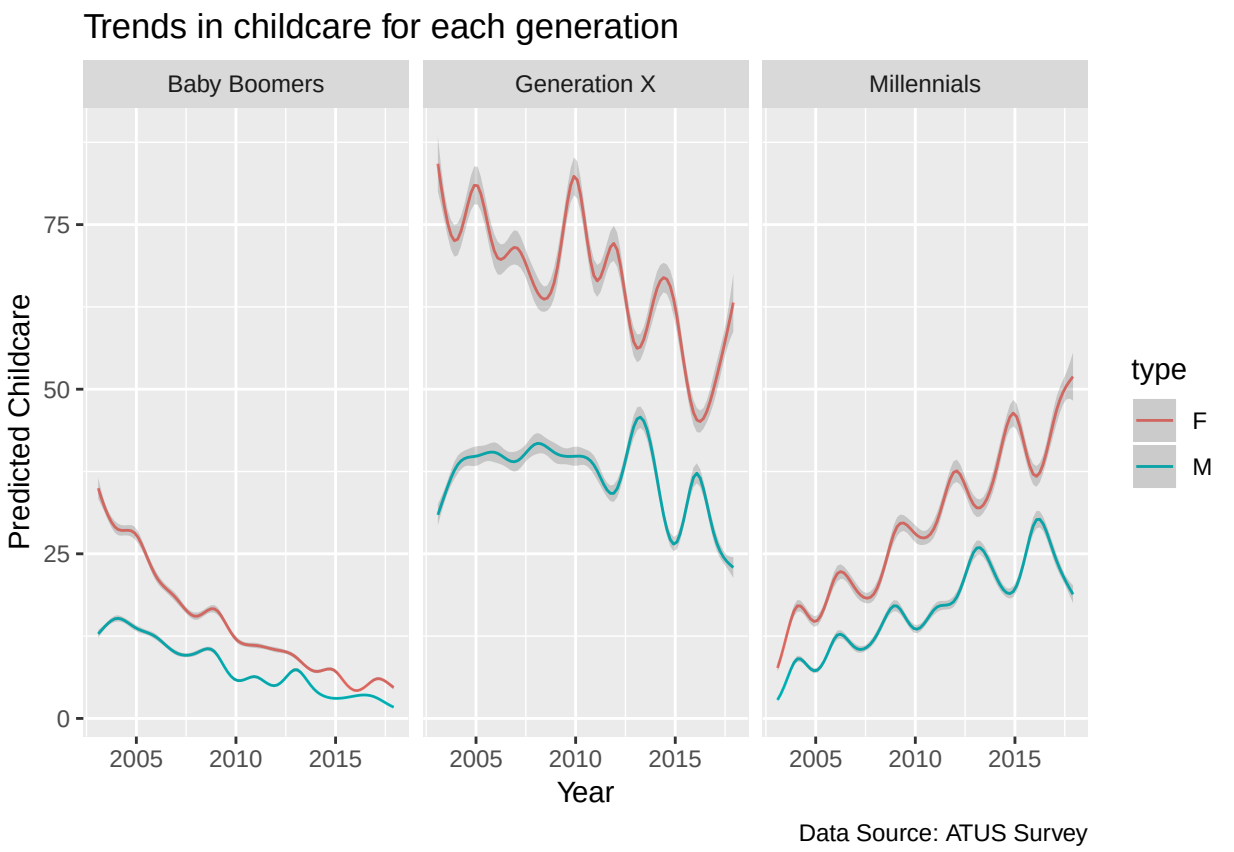

```
fit.models.and.plot(Gender.Roles.Train, 'Housework', 50)
```



```
fit.models.and.plot(Gender.Roles.Train, 'Food.Preparation', 60)
```



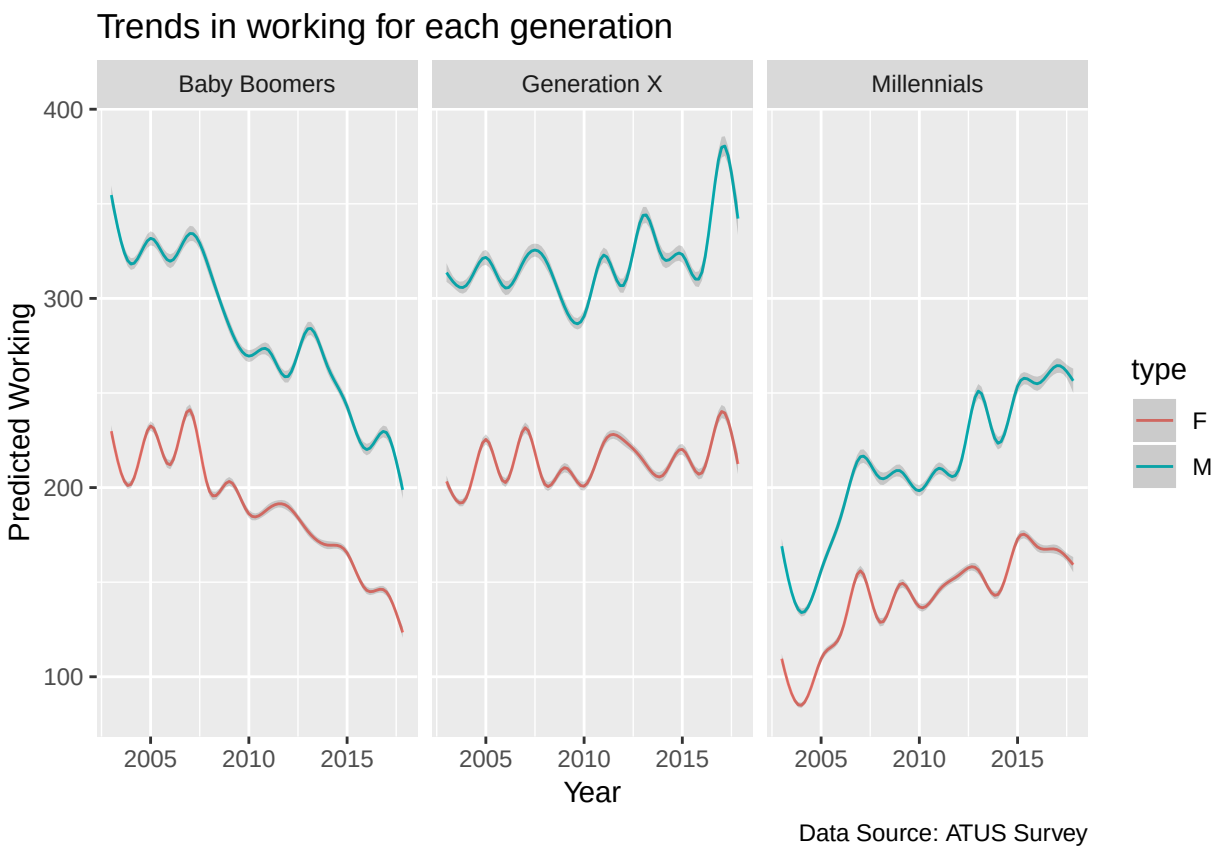

```
fit.models.and.plot(Gender.Roles.Train, 'Childcare', 50)
```



Generate plots to validate the trends observed above are consistent in the other half of the data. Further validation is carried out below but these provide some initial visual validation.

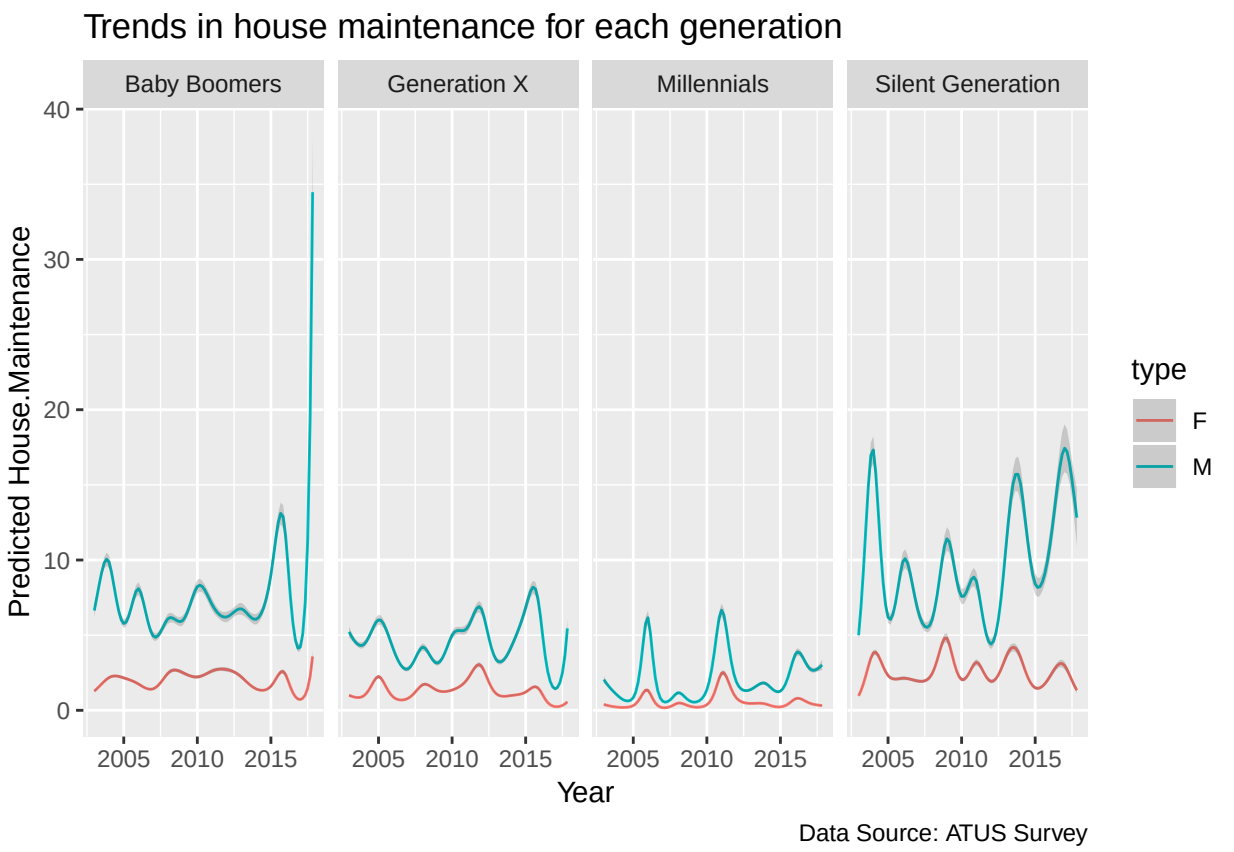
Validation and Presentation

```
fit.models.and.plot(Gender.Roles.Validate, 'Working', 250)
```



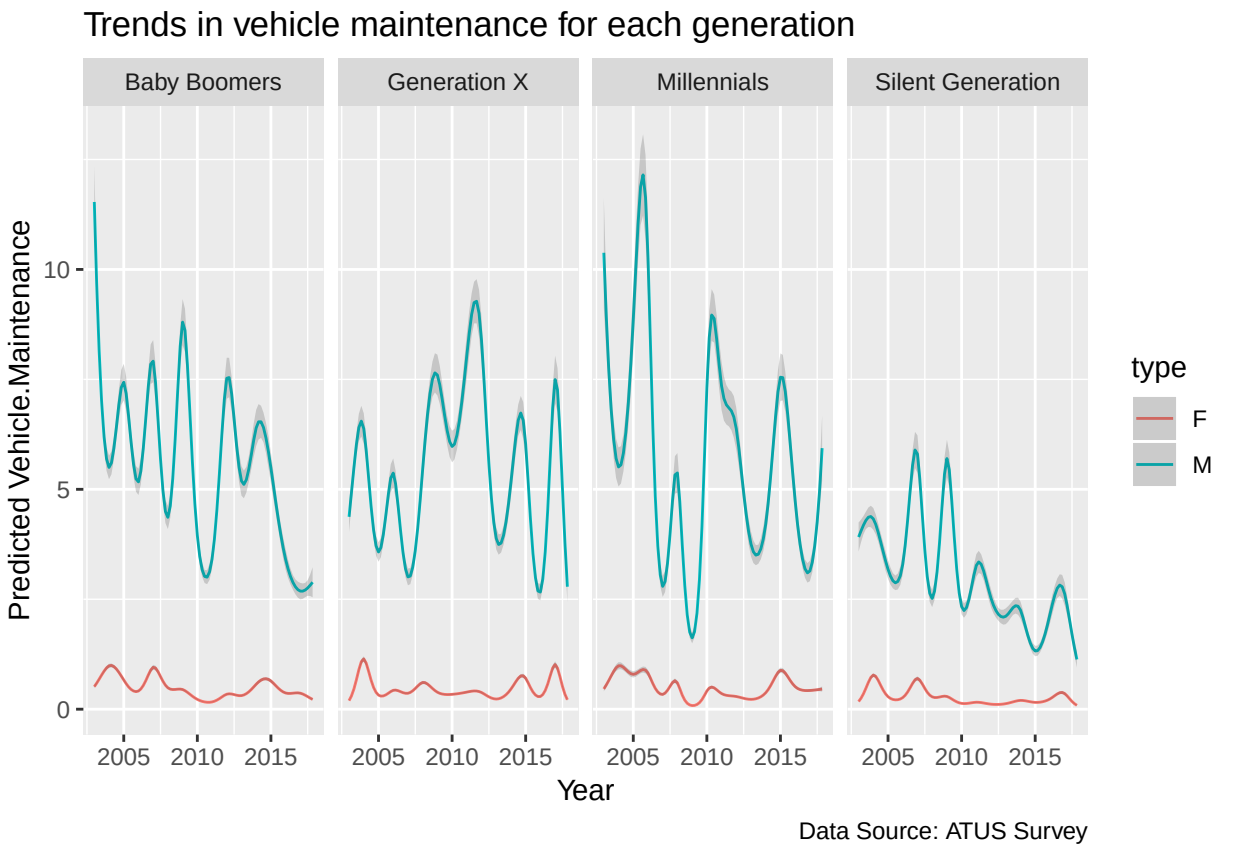

```
fit.models.and.plot(Gender.Roles.Validate, 'House.Maintenance', 10)
```

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

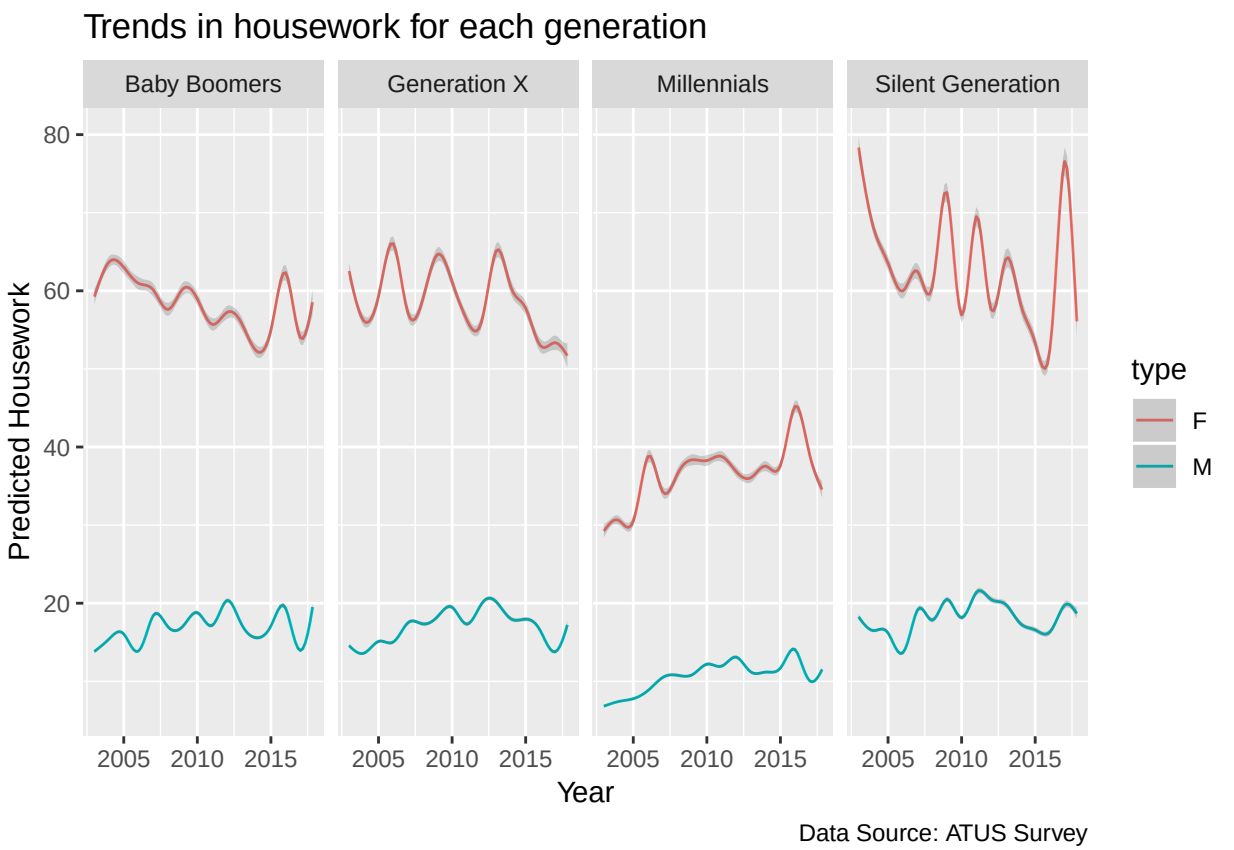



```
fit.models.and.plot(Gender.Roles.Validate, 'Vehicle.Maintenance', 10)
```

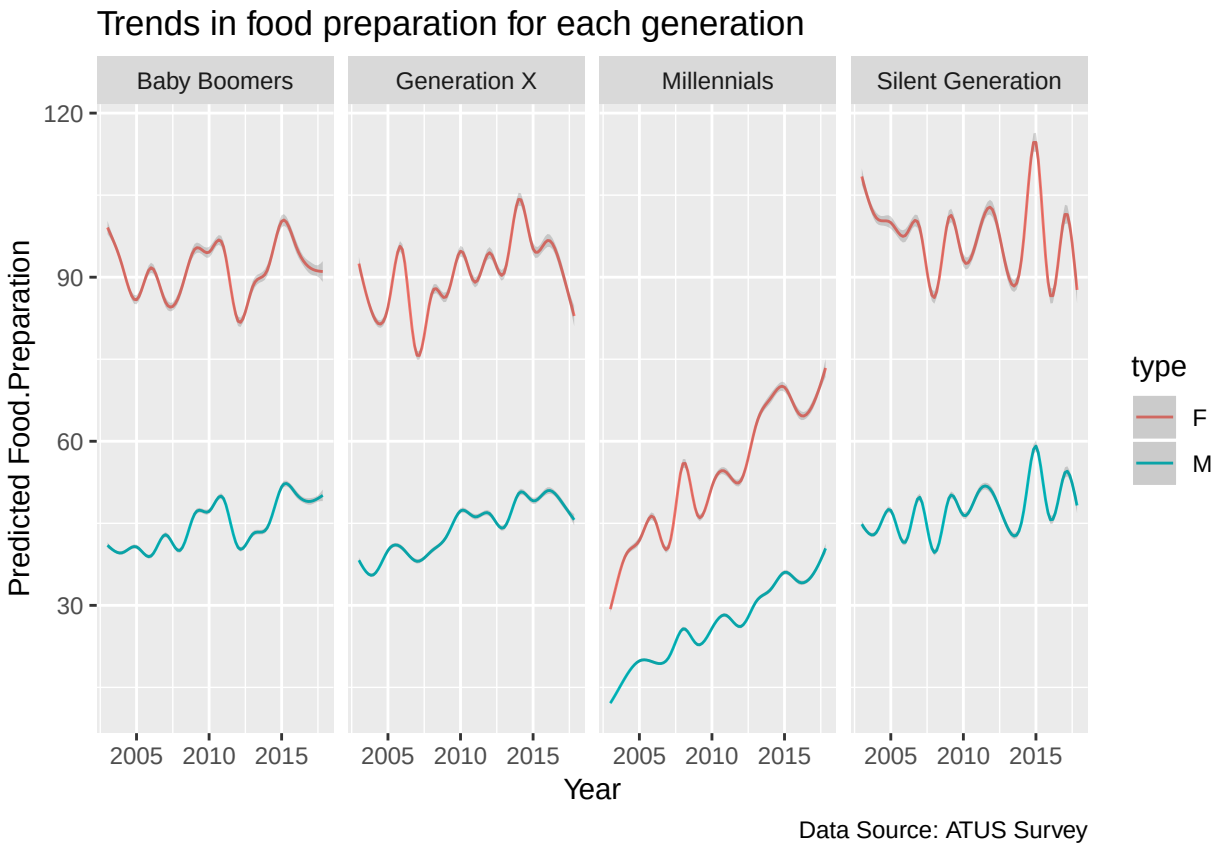
```
## Warning: glm.fit: algorithm did not converge
```



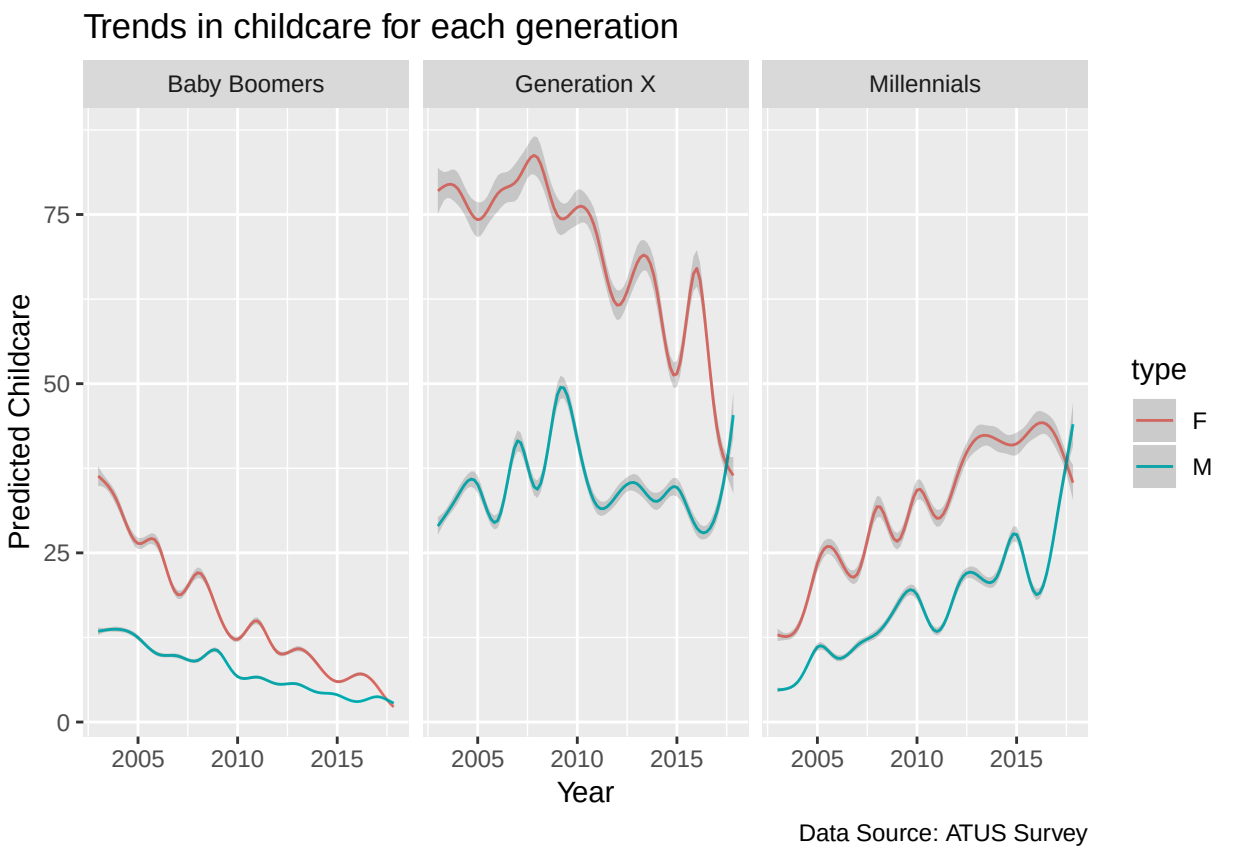

```
fit.models.and.plot(Gender.Roles.Validate, 'Housework', 50)
```



```
fit.models.and.plot(Gender.Roles.Validate, 'Food.Preparation', 60)
```

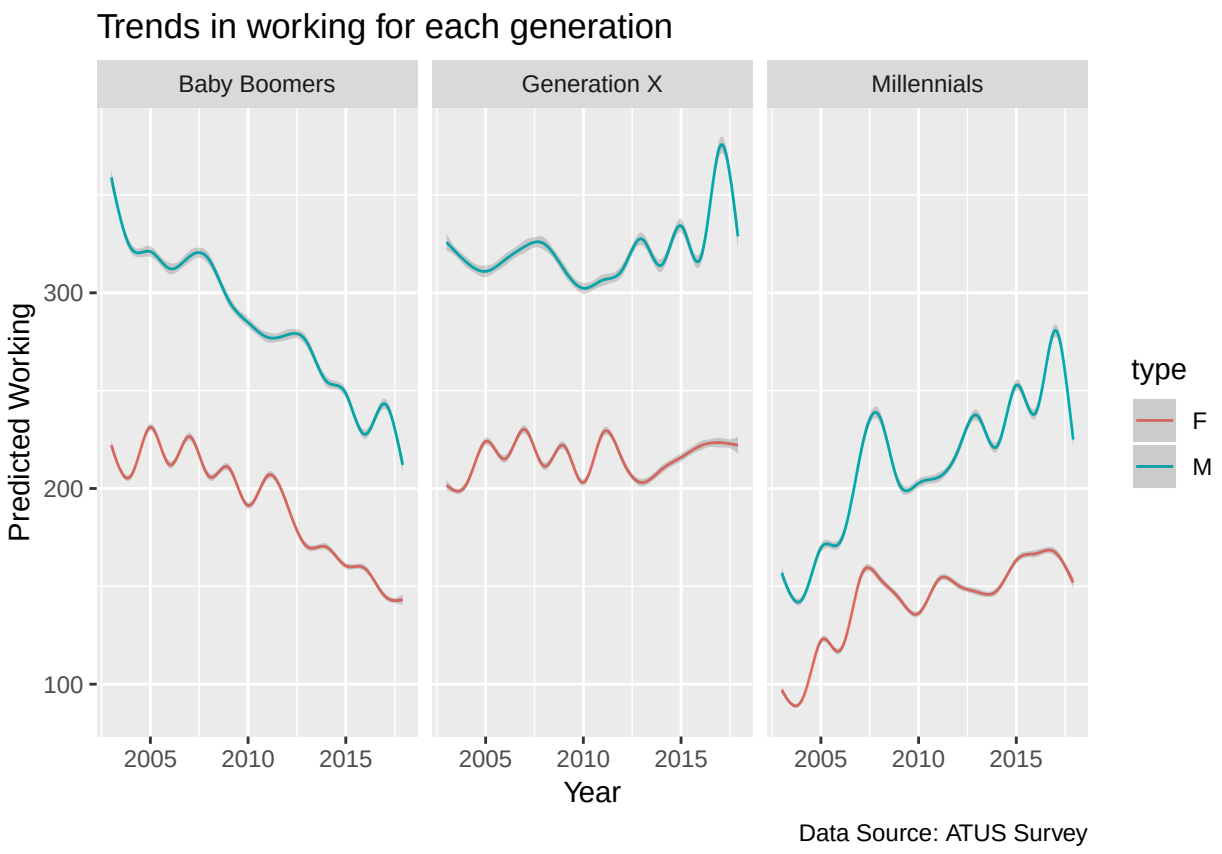



```
fit.models.and.plot(Gender.Roles.Validate, 'Childcare', 50)
```

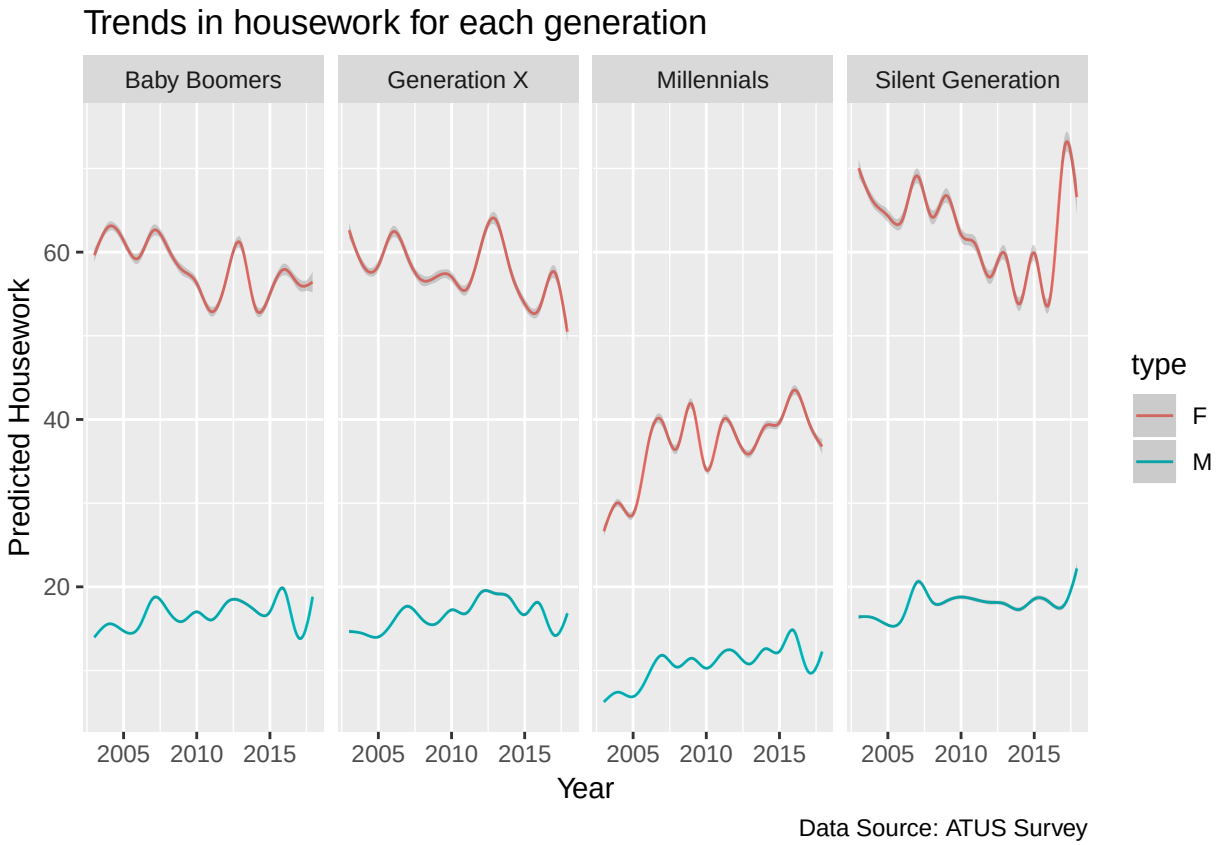


Generate plots on the entire dataset with the exception of July which is the directed month to exclude for this project. These plots are the ones included in the final report in order to conform with the requests of those who set the task. It is mentioned in the report that `House.Maintenance` and `Vehicle.Maintenance` are relatively uninteresting so no plots for these were included in the report to conserve space.

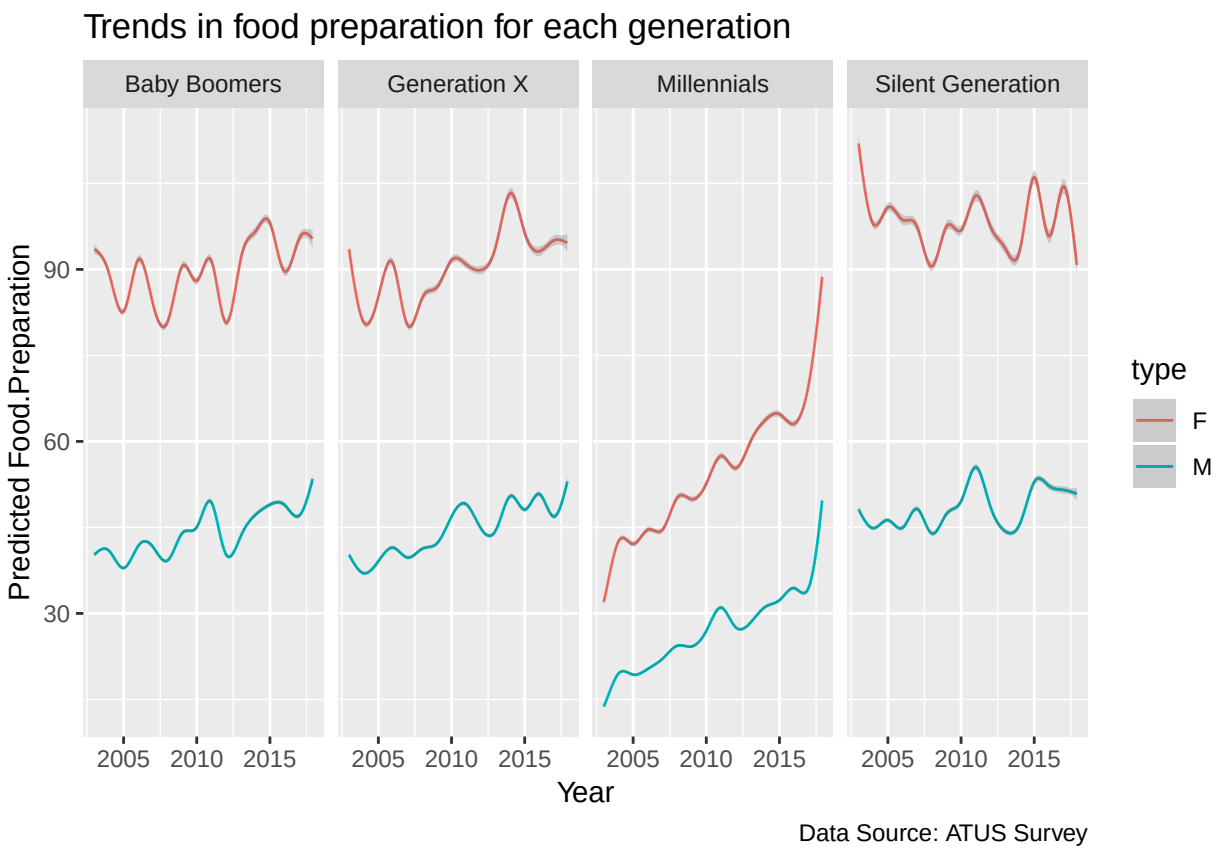
```
# Plot found in the Working section
fit.models.and.plot(Gender.Roles.Plotting, 'Working', 250)
```



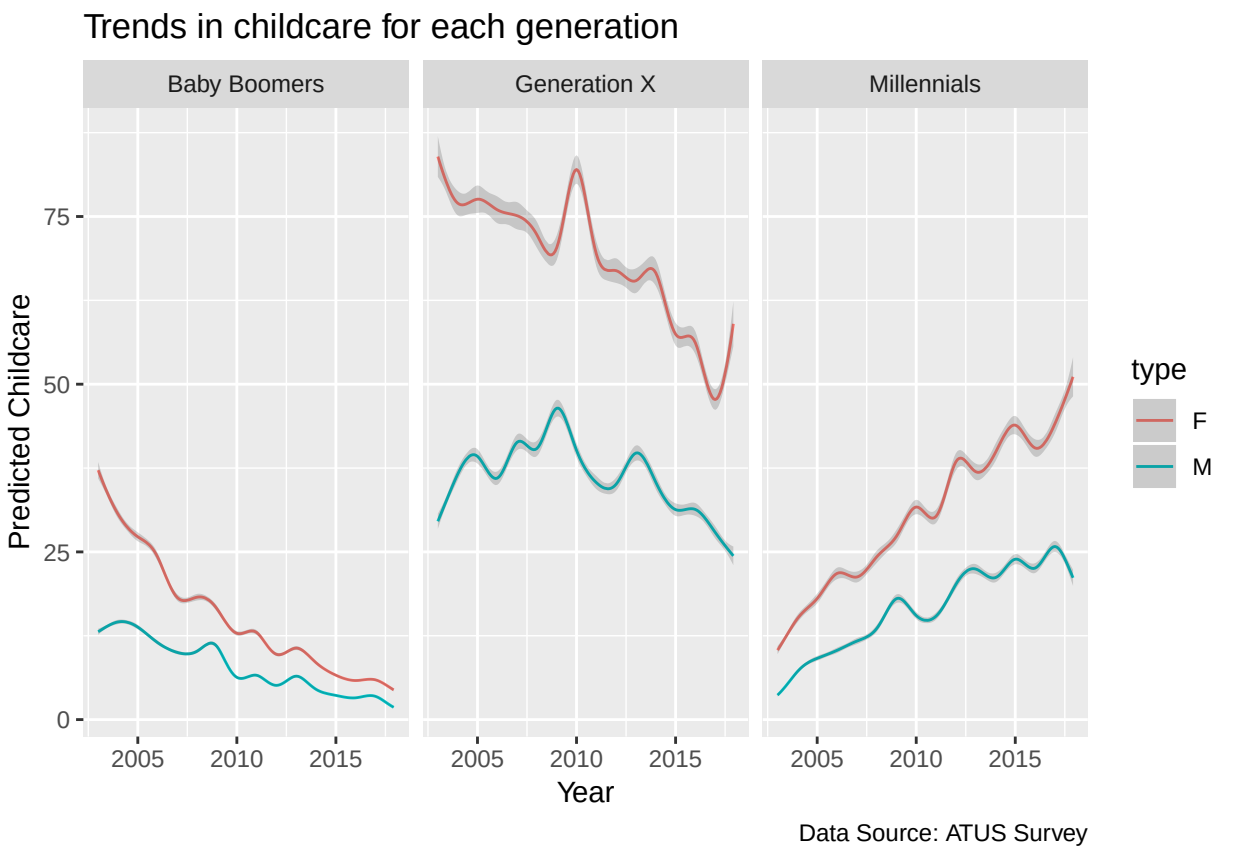

```
# Plot found in the Housework section
fit.models.and.plot(Gender.Roles.Plotting, 'Housework', 50)
```



```
# Plot found in the Food Preparation section
fit.models.and.plot(Gender.Roles.Plotting, 'Food.Preparation', 60)
```




```
# Plot found in the Childcare section
fit.models.and.plot(Gender.Roles.Plotting, 'Childcare', 50)
```



Create a `Gender.Roles.Differences` data frame which calculates the weighted means for each combination of `sex`, `variable` and `year` on a dataset only containing data from the defined validation months, these are then used to validate the hypotheses made earlier in the report. This process is quoted in the report but not included as it is potentially more statistical and complex than necessary for conveying the point of validation and t-tests / p-values behind said validation.

```
Gender.Roles.Validate.Dif <- filter(Gender.Roles.Data, month(TUDIARYDATE) %% 2 != 0)
Gender.Roles.Differences <- data.frame('Year' = c(), 'Variable' = c(), 'Difference' = c())
for (variable in c("Working", "House.Maintenance", "Vehicle.Maintenance", "Housework", "Food.Prep
    for (year in unique(Gender.Roles.Validate.Dif$TUYEAR)) {
        for (sex in c("M", "F")) {
            Filtered.Data <- filter(Gender.Roles.Validate.Dif, TESEX == sex, TUYEAR == year)
            bottom.sum <- sum(Filtered.Data$TUFNWGTP)
            top.sum <- sum(Filtered.Data$TUFNWGTP * Filtered.Data[,variable])
            weighted.values <- top.sum / bottom.sum
            if (sex == 'M') m.weighted.values <- weighted.values
        }
        weighted.difference <- m.weighted.values - weighted.values
        Gender.Roles.Differences <- rbind(Gender.Roles.Differences, data.frame('Year' = year, 'Va
    }
}
```

Code for running t-tests. the data set created above is filtered by variable (excluding `House.Maintenance` and `Vehicle.Maintenance` as these did not appear particularly interesting in the EDA as mentioned earlier) then a linear model is fitted along the datapoints, there is one point for each year. The summary for each model is then printed to allow for observation of the t-statistic and associated p-values. The `pt()` function can then be used on the coefficients extracted from the summary to conclude a one-tailed

significance t-test. Note that some values come back as 0.9... because of the negative differences, these results still align with our hypotheses and all of the p-values indicate significance.

```
for (variable in c("Working", "Housework", "Food.Preparation", "Childcare")) {
    print(variable)
    Filtered.Data <- filter(Gender.Roles.Differences, Variable == variable)
    model <- lm(data=Filtered.Data, Difference ~ Year)
    summary <- summary(model)
    print(summary)
    Year_t <- coefficients(summary)[2,3]
    lm_p_value <- pt(Year_t, model$df, lower=TRUE)
    print(lm_p_value)
}
```

```
## [1] "Working"
##
## Call:
## lm(formula = Difference ~ Year, data = Filtered.Data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.1021  -4.0418   0.2122   6.2561  10.8754
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2258.9133   993.9345   2.273   0.0407 *
## Year          -1.0869     0.4945  -2.198   0.0467 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.274 on 13 degrees of freedom
## Multiple R-squared:  0.2709, Adjusted R-squared:  0.2149
## F-statistic: 4.831 on 1 and 13 DF,  p-value: 0.04667
##
## [1] 0.02333554
## [1] "Housework"
##
## Call:
## lm(formula = Difference ~ Year, data = Filtered.Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5212 -1.7565 -0.2742  1.2925  4.2007
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1534.0124   260.2373  -5.895 5.28e-05 ***
## Year            0.7439     0.1295   5.746 6.76e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.166 on 13 degrees of freedom
## Multiple R-squared:  0.7175, Adjusted R-squared:  0.6957
## F-statistic: 33.01 on 1 and 13 DF,  p-value: 6.756e-05
##
## [1] 0.9999662
## [1] "Food.Preparation"
##
## Call:
## lm(formula = Difference ~ Year, data = Filtered.Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2021 -1.6120 -0.1631  1.1855  5.0499
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1009.4536   289.7418  -3.484  0.00404 **
## Year            0.4807     0.1441   3.335  0.00538 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.412 on 13 degrees of freedom
## Multiple R-squared:  0.461,  Adjusted R-squared:  0.4195
## F-statistic: 11.12 on 1 and 13 DF,  p-value: 0.005378
##
## [1] 0.9973109
## [1] "Childcare"
##
## Call:
## lm(formula = Difference ~ Year, data = Filtered.Data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.4468 -0.9693  0.5511  1.2091  4.1373
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -675.1851   282.9403  -2.386   0.0329 *
## Year           0.3265     0.1408   2.320   0.0373 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.355 on 13 degrees of freedom
## Multiple R-squared:  0.2928, Adjusted R-squared:  0.2384
## F-statistic: 5.381 on 1 and 13 DF,  p-value: 0.03726
##
## [1] 0.9813679
```

Map Plot

References

[1] Bureau of Labor Statistics, "The american time use survey." https://www.bls.gov/tus/, 2017.