```
if (!requireNamespace("lmtest", quietly = TRUE)) install.packages("lmtest")
if (!requireNamespace("dplyr", quietly = TRUE)) install.packages("dplyr")
```

Question 1: Create a multivariate time series; perform any interpolations.

Load Packages

```
#
# install.packages("haven")
# install.packages("devtools", dependencies = TRUE)
# install.packages("car")
# install.packages("ggplot2")
# install.packages("plyr")
# install.packages("forecast")
# install.packages("fUnitRoots")
# install.packages("tseries")


#load packages
library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.4.2
```

```
library(plyr)
```

```
Warning: package 'plyr' was built under R version 4.4.2
```

```
library(lmtest)
```

```
Warning: package 'lmtest' was built under R version 4.4.2
```

```
Loading required package: zoo
```

```
Attaching package: 'zoo'
```

```
The following objects are masked from 'package:base':

    as.Date, as.Date.numeric
```

```
library(car)
```

```
Warning: package 'car' was built under R version 4.4.2

Loading required package: carData

Warning: package 'carData' was built under R version 4.4.2
```

```r
library(dplyr)
```

```
Warning: package 'dplyr' was built under R version 4.4.2


Attaching package: 'dplyr'

The following object is masked from 'package:car':

    recode

The following objects are masked from 'package:plyr':

    arrange, count, desc, failwith, id, mutate, rename, summarise,
    summarize

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```r
library(haven)
```

```
Warning: package 'haven' was built under R version 4.4.2
```

```r
library(forecast)
```

```
Warning: package 'forecast' was built under R version 4.4.2

Registered S3 method overwritten by 'quantmod':
  method            from
  as.zoo.data.frame zoo
```

```r
library(fUnitRoots)
```

```
Warning: package 'fUnitRoots' was built under R version 4.4.2
```

```r
library(tseries)
```

```
Warning: package 'tseries' was built under R version 4.4.2
```

Load the Data set

```
# Load the haven package
library(haven)



# Load the Stata file
file_path <- "C:/Users/nmv2125/Downloads/GSS_stata (2)/GSS_stata/gss7222_r4.dta"
gss_data <- read_dta(file_path)

# View the first few rows
head(gss_data)
```

```
# A tibble: 6 × 6,696
  year     id wrkstat hrs1        hrs2        evwork      occ   prestige wrkslf
  <dbl> <dbl> <dbl+l> <dbl+lbl>   <dbl+lbl>   <dbl+lbl>   <dbl> <dbl+lb> <dbl+l>
1 1972      1 1 [wor… NA(i) [iap] NA(i) [iap] NA(i) [iap] 205   50       2 [som…
2 1972      2 5 [ret… NA(i) [iap] NA(i) [iap]     1 [yes] 441   45       2 [som…
3 1972      3 2 [wor… NA(i) [iap] NA(i) [iap] NA(i) [iap] 270   44       2 [som…
4 1972      4 1 [wor… NA(i) [iap] NA(i) [iap] NA(i) [iap]   1   57       2 [som…
5 1972      5 7 [kee… NA(i) [iap] NA(i) [iap]     1 [yes] 385   40       2 [som…
6 1972      6 1 [wor… NA(i) [iap] NA(i) [iap] NA(i) [iap] 281   49       2 [som…
# ℹ 6,687 more variables: wrkgovt <dbl+lbl>, commute <dbl+lbl>,
#   industry <dbl+lbl>, occ80 <dbl+lbl>, prestg80 <dbl+lbl>, indus80 <dbl+lbl>,
#   indus07 <dbl+lbl>, occonet <dbl+lbl>, found <dbl+lbl>, occ10 <dbl+lbl>,
#   occindv <dbl+lbl>, occstatus <dbl+lbl>, occtag <dbl+lbl>,
#   prestg10 <dbl+lbl>, prestg105plus <dbl+lbl>, indus10 <dbl+lbl>,
#   indstatus <dbl+lbl>, indtag <dbl+lbl>, marital <dbl+lbl>,
#   martype <dbl+lbl>, agewed <dbl+lbl>, divorce <dbl+lbl>, …
```

```
# Check structure of the dataset
#str(gss_data)
```

Here are the variables I've chosen: fefam, educ, happy7, sprtprsn, discaffw

```
# print(gss_data$fefam)
# print(gss_data$educ)
# print(gss_data$happy7)
# print(gss_data$sprtprsn)
# print(gss_data$discaffw)
```

Recode Variables to Prepare for Dataset

```
# Subset only the relevant variables
vars <- c("year", "fefam", "educ", "happy7", "sprtprsn", "discaffw")
sub <- gss_data[, vars]

# Recode variables: turn categorical responses into meaningful numeric ones
sub <- sub %>%
  mutate(
    fefam_bin = ifelse(fefam == 1, 1, 0),  # "strongly agree" as 1, others as 0
    educ_years = as.numeric(educ),         # education years already numeric
    happy_score = case_when(               # reverse code happiness
      happy7 == 1 ~ 7,  # completely happy = 7
      happy7 == 2 ~ 6,
      happy7 == 3 ~ 5,
      happy7 == 4 ~ 4,
      happy7 == 5 ~ 3,
      happy7 == 6 ~ 2,
      happy7 == 7 ~ 1,  # completely unhappy = 1
      TRUE ~ NA_real_
    ),
    spiritual_score = case_when(
      sprtprsn == 1 ~ 4,  # very spiritual = 4
      sprtprsn == 2 ~ 3,
      sprtprsn == 3 ~ 2,
      sprtprsn == 4 ~ 1,  # not spiritual = 1
      TRUE ~ NA_real_
    ),
    discaffw_likely = case_when(
      discaffw == 1 ~ 4,  # very likely = 4
      discaffw == 2 ~ 3,
      discaffw == 3 ~ 2,
      discaffw == 4 ~ 1,  # very unlikely = 1
      TRUE ~ NA_real_
    )
  )
```

Aggregate Data by Year

```
# Group data by year and calculate means, ignoring NA values
by_year <- sub %>%
  group_by(year) %>%
  summarise(
    fefam_pct = mean(fefam_bin, na.rm = TRUE) * 100,  # Percentage who "strongly agree"
    avg_educ = mean(educ_years, na.rm = TRUE),         # Average education years
    avg_happy = mean(happy_score, na.rm = TRUE),       # Average happiness score
    avg_spiritual = mean(spiritual_score, na.rm = TRUE),  # Avg spirituality
    avg_discaffw = mean(discaffw_likely, na.rm = TRUE)    # Avg likelihood of discrimination
  )
```

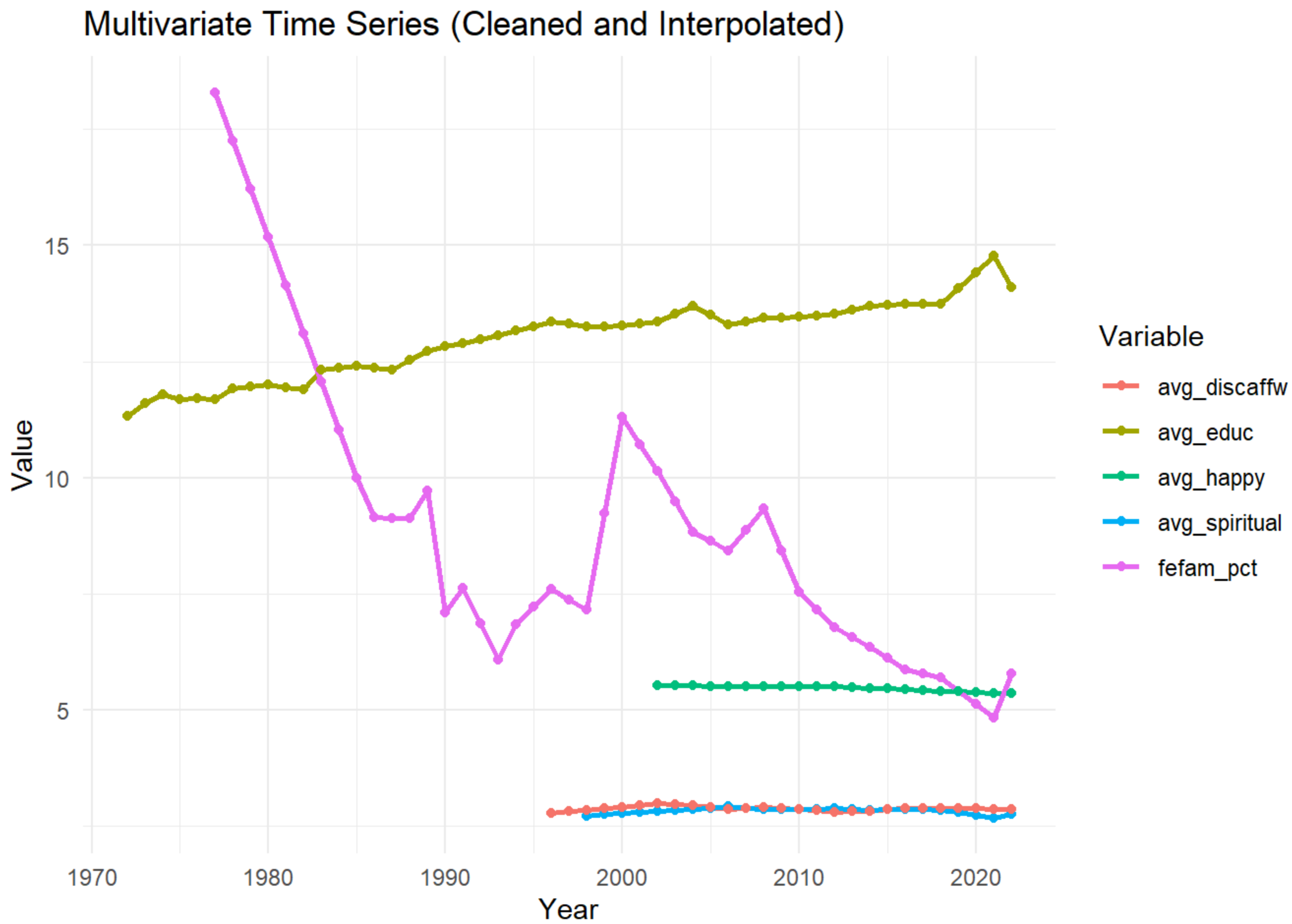
Interpolate Missing Years

```
# Add missing years explicitly if required
all_years <- data.frame(year = seq(min(by_year$year), max(by_year$year), by = 1))
by_year <- full_join(by_year, all_years, by = "year") %>%
  arrange(year)

# Interpolate missing values for smoother time series
by_year_interp <- by_year %>%
  mutate(
    fefam_pct = na.approx(fefam_pct, na.rm = FALSE),
    avg_educ = na.approx(avg_educ, na.rm = FALSE),
    avg_happy = na.approx(avg_happy, na.rm = FALSE),
    avg_spiritual = na.approx(avg_spiritual, na.rm = FALSE),
    avg_discaffw = na.approx(avg_discaffw, na.rm = FALSE)
  )
```

Visualize Clean Data

```
# Convert to long format for easier plotting
library(tidyr)
by_year_long <- pivot_longer(by_year_interp, cols = -year, names_to = "variable", values_to = "va

# Plot the time series for all variables
ggplot(by_year_long, aes(x = year, y = value, color = variable)) +
  geom_line(linewidth = 1) +  # Updated size aesthetic
  geom_point() +
  labs(title = "Multivariate Time Series (Cleaned and Interpolated)",
       x = "Year", y = "Value", color = "Variable") +
  theme_minimal()
```

```
Warning: Removed 85 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 85 rows containing missing values or values outside the scale range
(`geom_point()`).
```



Question 2: Graph the relationships between X and Y. Explain how you think Y should relate to your key Xs.

Melt the Time Series:

reshape the cleaned time series data into a long format. This makes it easy to plot different variables together.

```
library(reshape2)
```

```
Warning: package 'reshape2' was built under R version 4.4.2
```

```
Attaching package: 'reshape2'
```

```
The following object is masked from 'package:tidyr':

    smiths
```

```
# Melt the data into long format for plotting
melt_my_ts <- function(ts_data, time_var, keep_vars) {
  # ts_data: data.frame of my time series
  # time_var: name of the time column
  # keep_vars: variables to keep for plotting

  # Ensure time variable is in keep.vars
  if (!(time_var %in% keep_vars)) {
    keep_vars <- c(keep_vars, time_var)
  }

  melted_data <- ts_data[, keep_vars]
  melted_data <- melt(melted_data, id.vars = time_var)
  colnames(melted_data)[which(colnames(melted_data) == time_var)] <- "time"
  return(melted_data)
}

# Variables to plot
keep_vars <- c("year", "fefam_pct", "avg_educ", "avg_happy", "avg_spiritual", "avg_discaffw")

# Melt the data
plot_data <- melt_my_ts(by_year_interp, time_var = "year", keep_vars = keep_vars)
head(plot_data)
```

```
  time  variable    value
1 1972 fefam_pct       NA
2 1973 fefam_pct       NA
3 1974 fefam_pct       NA
4 1975 fefam_pct       NA
5 1976 fefam_pct       NA
6 1977 fefam_pct 18.29674
```

Define a Plotting Function

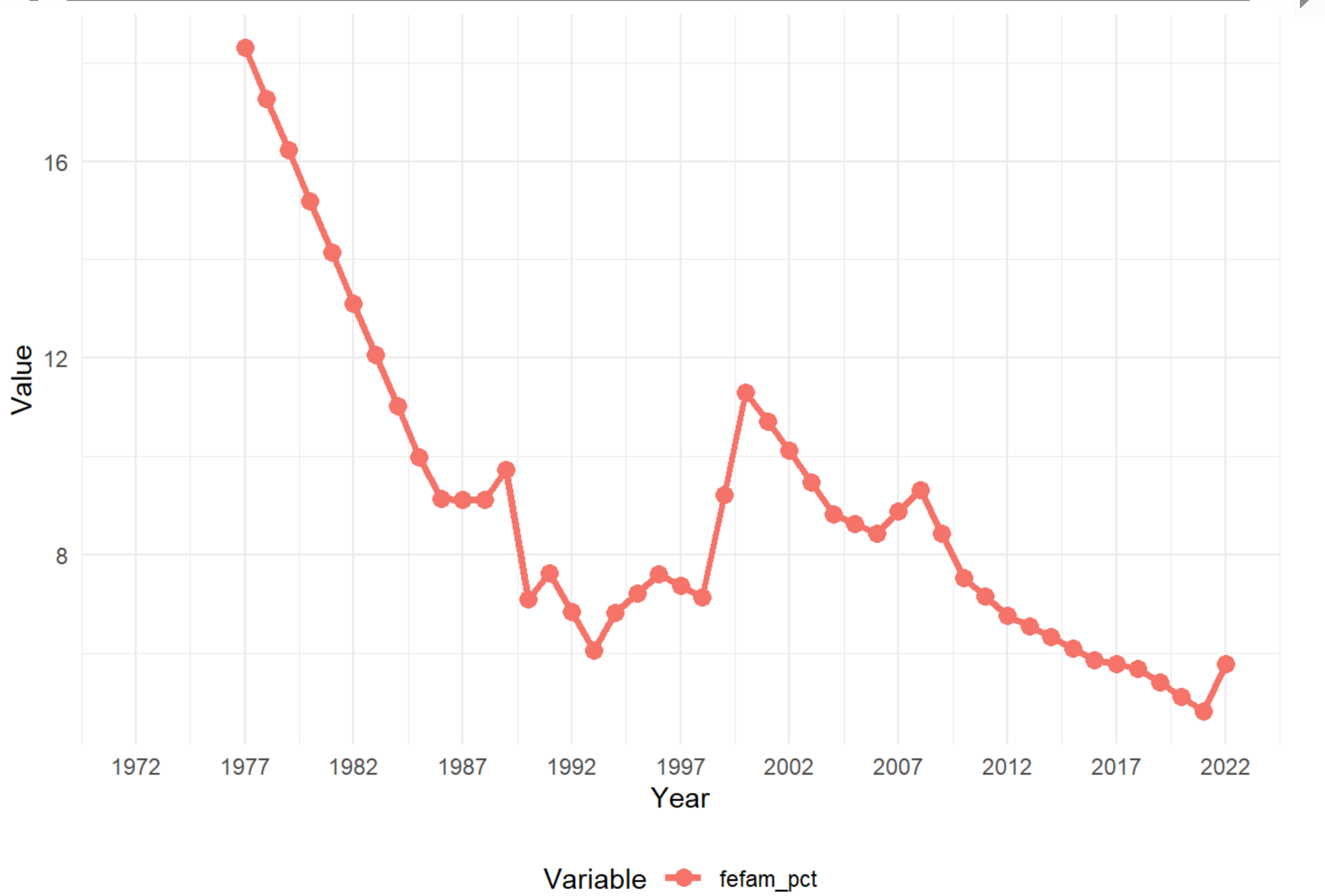
```
library(ggplot2)

plot_my_ts <- function(data, varlist, line = TRUE, point = TRUE, pointsize = 3, linewidth = 1.25)
  # data: melted data frame
  # varlist: character vector of variables to plot
  if (missing(varlist)) {
    gg <- ggplot(data, aes(x = time, y = value, color = variable))
  } else {
    gg <- ggplot(data[data$variable %in% varlist, ], aes(x = time, y = value, color = variable))
  }

  if (line) gg <- gg + geom_line(linewidth = linewidth)
  if (point) gg <- gg + geom_point(size = pointsize)
```

```
  gg <- gg + labs(x = "Year", y = "Value", color = "Variable") +
    theme_minimal() +
    theme(legend.position = "bottom") +
    scale_x_continuous(breaks = seq(min(data$time), max(data$time), by = 5))

  return(gg)
}
```

Generate Plot for Relationships Between X & Y

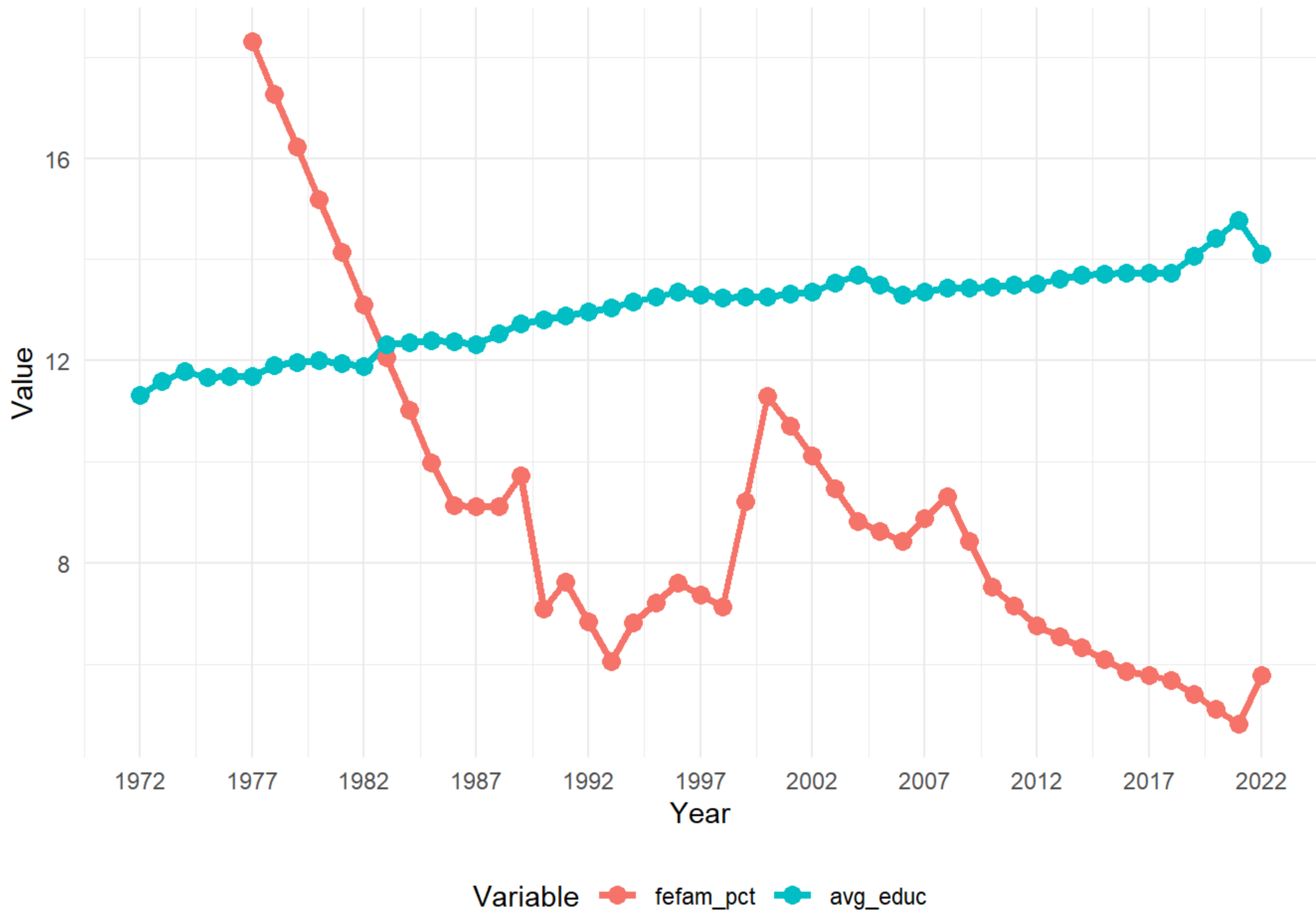
```
# Plot fefam_pct over time (Y variable)
plot_fefam <- plot_my_ts(plot_data, varlist = c("fefam_pct"))
plot_fefam
```

```
Warning: Removed 5 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 5 rows containing missing values or values outside the scale range
(`geom_point()`).
```



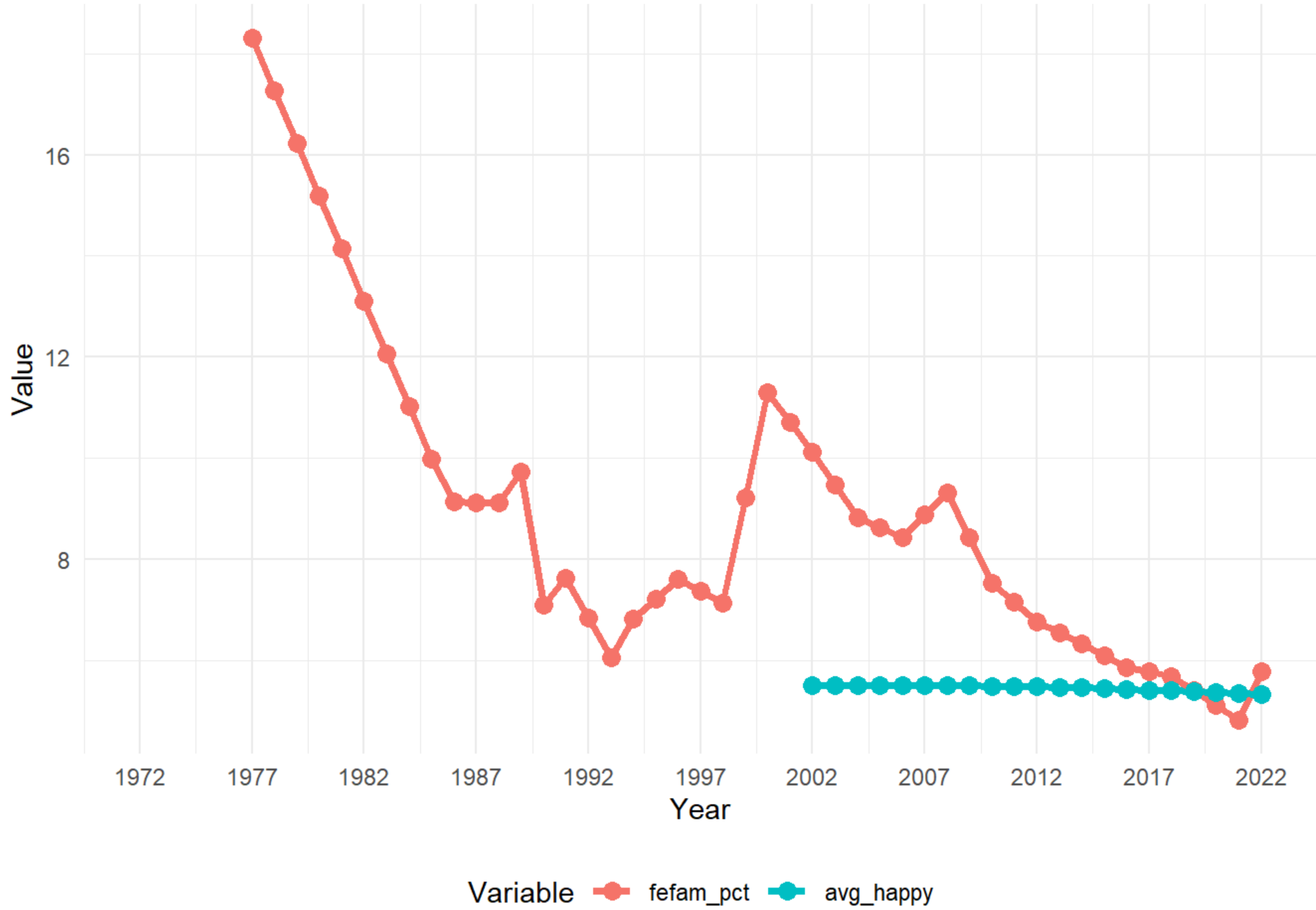
```
# Plot avg_educ (X) and fefam_pct (Y)
plot_educ <- plot_my_ts(plot_data, varlist = c("fefam_pct", "avg_educ"))
plot_educ
```

```
Warning: Removed 5 rows containing missing values or values outside the scale range
(`geom_line()`).
Removed 5 rows containing missing values or values outside the scale range
(`geom_point()`).
```



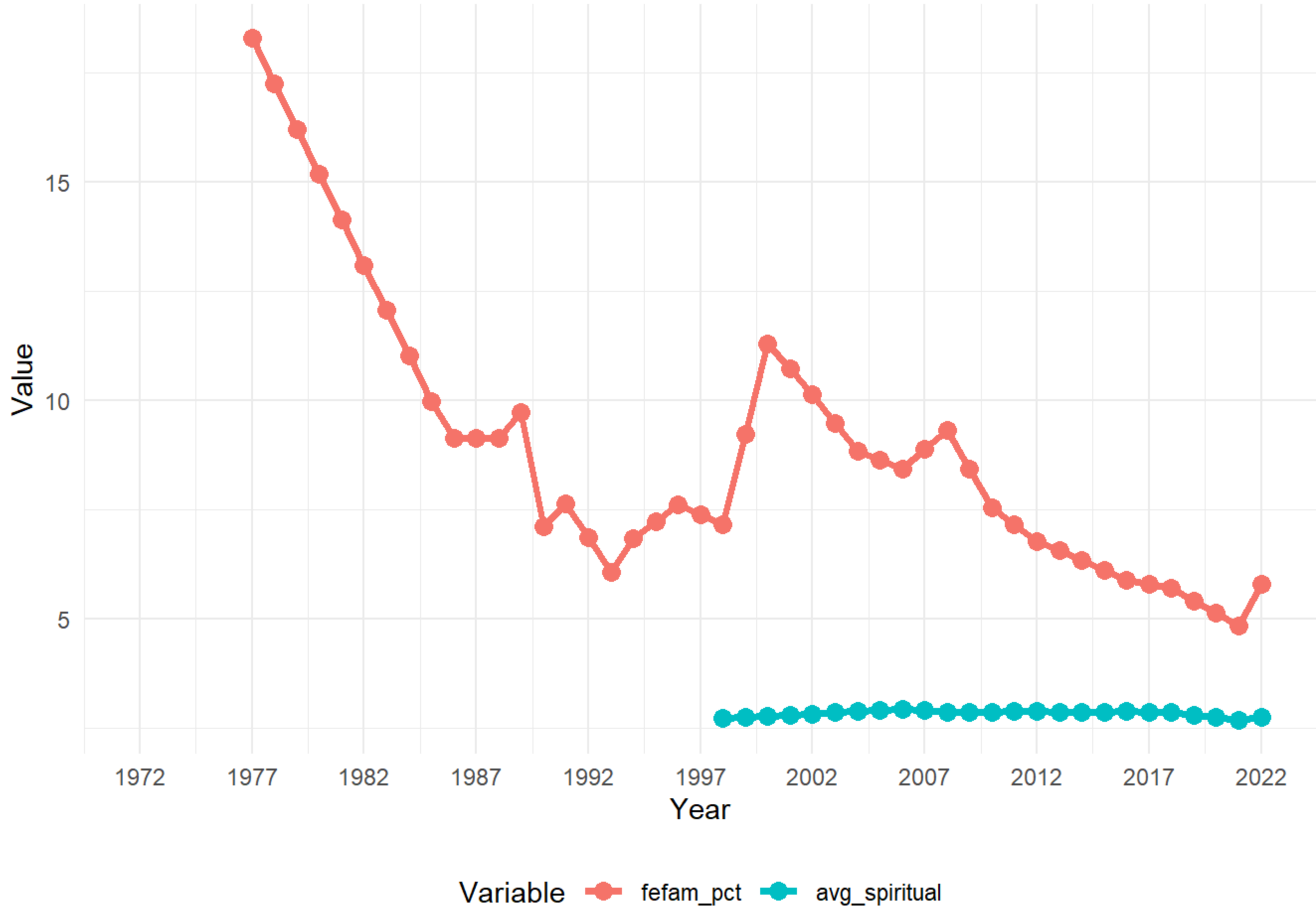

```
# Plot avg_happy (X) and fefam_pct (Y)
plot_happy <- plot_my_ts(plot_data, varlist = c("fefam_pct", "avg_happy"))
plot_happy
```

```
Warning: Removed 35 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 35 rows containing missing values or values outside the scale range
(`geom_point()`).
```



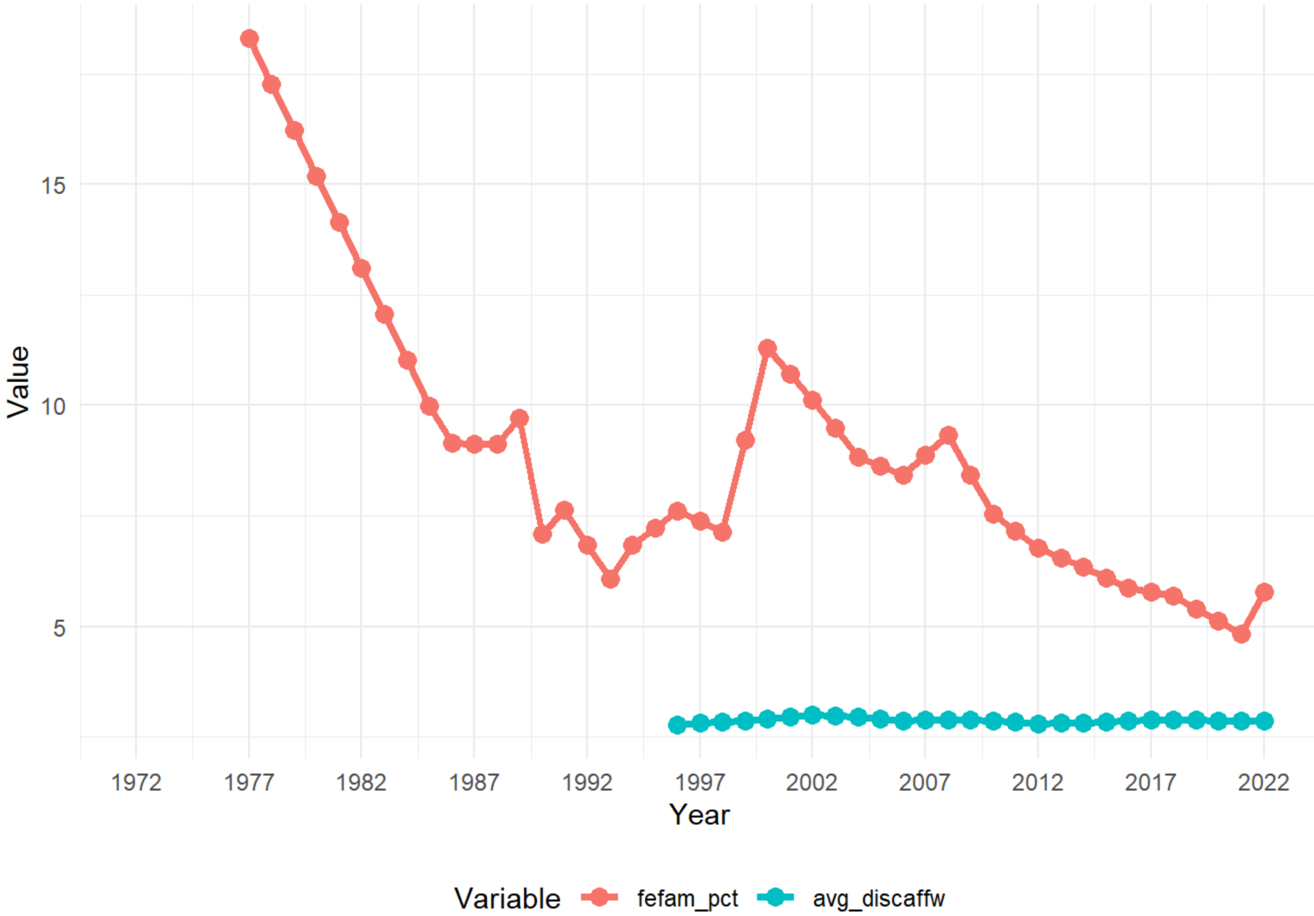

```
# Plot avg_spiritual (X) and fefam_pct (Y)
plot_spiritual <- plot_my_ts(plot_data, varlist = c("fefam_pct", "avg_spiritual"))
plot_spiritual
```

```
Warning: Removed 31 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 31 rows containing missing values or values outside the scale range
(`geom_point()`).
```




```
# Plot avg_discaffw (X) and fefam_pct (Y)
plot_discaffw <- plot_my_ts(plot_data, varlist = c("fefam_pct", "avg_discaffw"))
plot_discaffw
```

```
Warning: Removed 29 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 29 rows containing missing values or values outside the scale range
(`geom_point()`).
```



- `fefam_pct` **vs** `avg_educ`: kind of **negative relationship**—higher education could correlate with less agreement to traditional gender roles.
- `fefam_pct` **vs** `avg_happy`: Happiness have a neutral relationship, depending on individual perspectives.
- `fefam_pct` **vs** `avg_spiritual`: I thought higher spirituality might correlate with higher agreement to traditional family roles, or lower agreement to traditional family roles, depending on the religious and spirituality values. Turns out, spirituality is quiet neutral
- `fefam_pct` **vs** `avg_discaffw`: I thought if people perceive greater workplace discrimination against women (`discaffw`), agreement to traditional roles might also increase, however, it seems neutral and there doesn't seem to be a relationship

Question 3: Run a simple time series regression, with one X and no trend. Interpret it.

Simple Time Series Regression

```
# Simple time series regression: fefam_pct ~ avg_educ
lm_fefam <- lm(fefam_pct ~ avg_educ, data = by_year_interp)
```

```r
# Summary of the regression
summary(lm_fefam)
```

```
Call:
lm(formula = fefam_pct ~ avg_educ, data = by_year_interp)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.1595 -0.9548 -0.4269  1.0700  3.9326 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  58.3565     5.0702  11.510 7.35e-15 ***
avg_educ     -3.7638     0.3855  -9.763 1.39e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.83 on 44 degrees of freedom
  (5 observations deleted due to missingness)
Multiple R-squared:  0.6842,	Adjusted R-squared:  0.677 
F-statistic: 95.31 on 1 and 44 DF,  p-value: 1.394e-12
```

Slope (avg_educ): -3.7638

- For every **one-unit increase** in average education (`avg_educ`), the percentage agreeing with traditional family roles (`fefam_pct`) **decreases by 3.76 percentage points**.
- The negative coefficient suggests that **higher education levels are associated with a decline in agreement** with traditional family norms.

P-values:

- Both the **Intercept** and `avg_educ` are highly significant (`p < 0.001`), as seen by the `***` symbols.
- This means there is very strong evidence that average education (`avg_educ`) influences `fefam_pct`.

Test for Heteroskedascity

```r
# Install and load lmtest for Breusch-Pagan Test
install.packages("lmtest")
```

```
Warning: package 'lmtest' is in use and will not be installed
```

```r
library(lmtest)

# Test for heteroskedasticity
bptest(lm_fefam)
```

```
	studentized Breusch-Pagan test

data:  lm_fefam
BP = 10.235, df = 1, p-value = 0.001378
```

The Breusch-Pagan test checks whether the variance of the residuals (errors) is constant or changes systematically with the predictor variable. Non-constant variance is called heteroskedasticity, which can affect the validity of regression results.

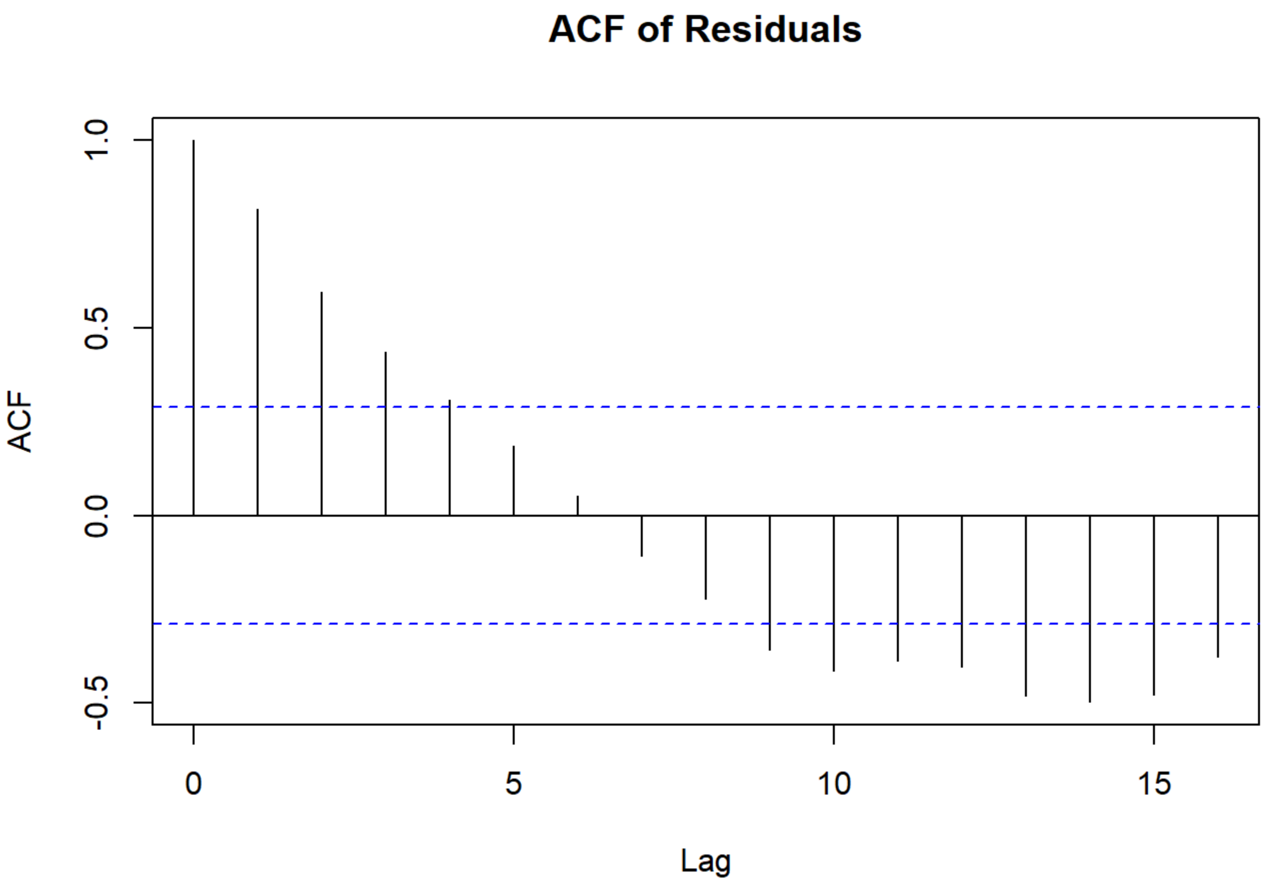
`studentized Breusch-Pagan test  data:  lm_fefam BP = 10.235, df = 1, p-value = 0.001378`

- **BP statistic = 10.235**
  - This is the test statistic for the Breusch-Pagan test. A larger value indicates more evidence of heteroskedasticity.
- **Degrees of freedom (df) = 1**
  - This corresponds to the single predictor variable `avg_educ` in my model.
- **p-value = 0.001378**
  - Since the p-value is **less than 0.05**, we **reject the null hypothesis** of homoskedasticity (constant variance).

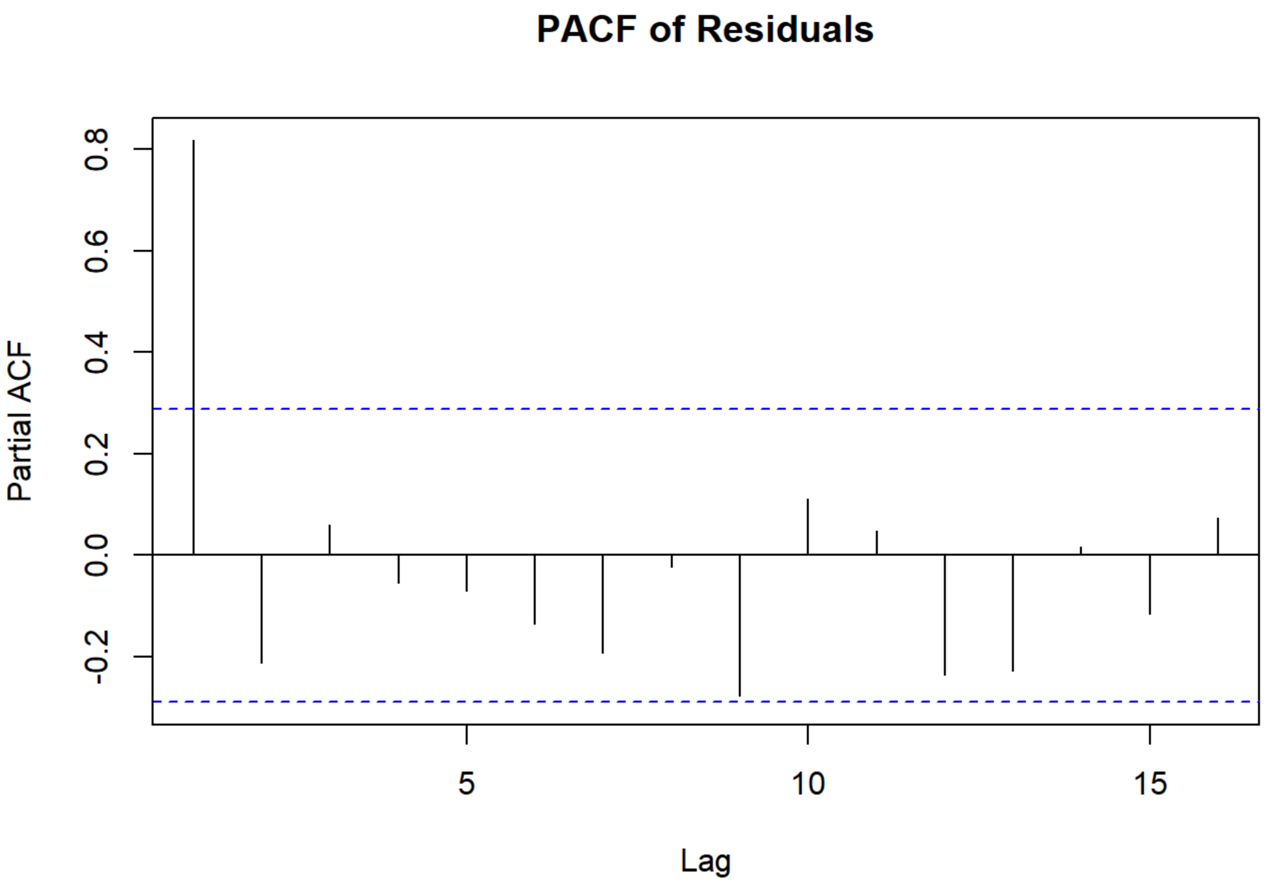
The residuals' variance is not constant and is likely related to avg_educ.

Checking for Autocorrelation in Residuals

```r
# Extract residuals
resid_fefam <- lm_fefam$residuals

# Autocorrelation Function (ACF) plot
acf(resid_fefam, main = "ACF of Residuals")
```



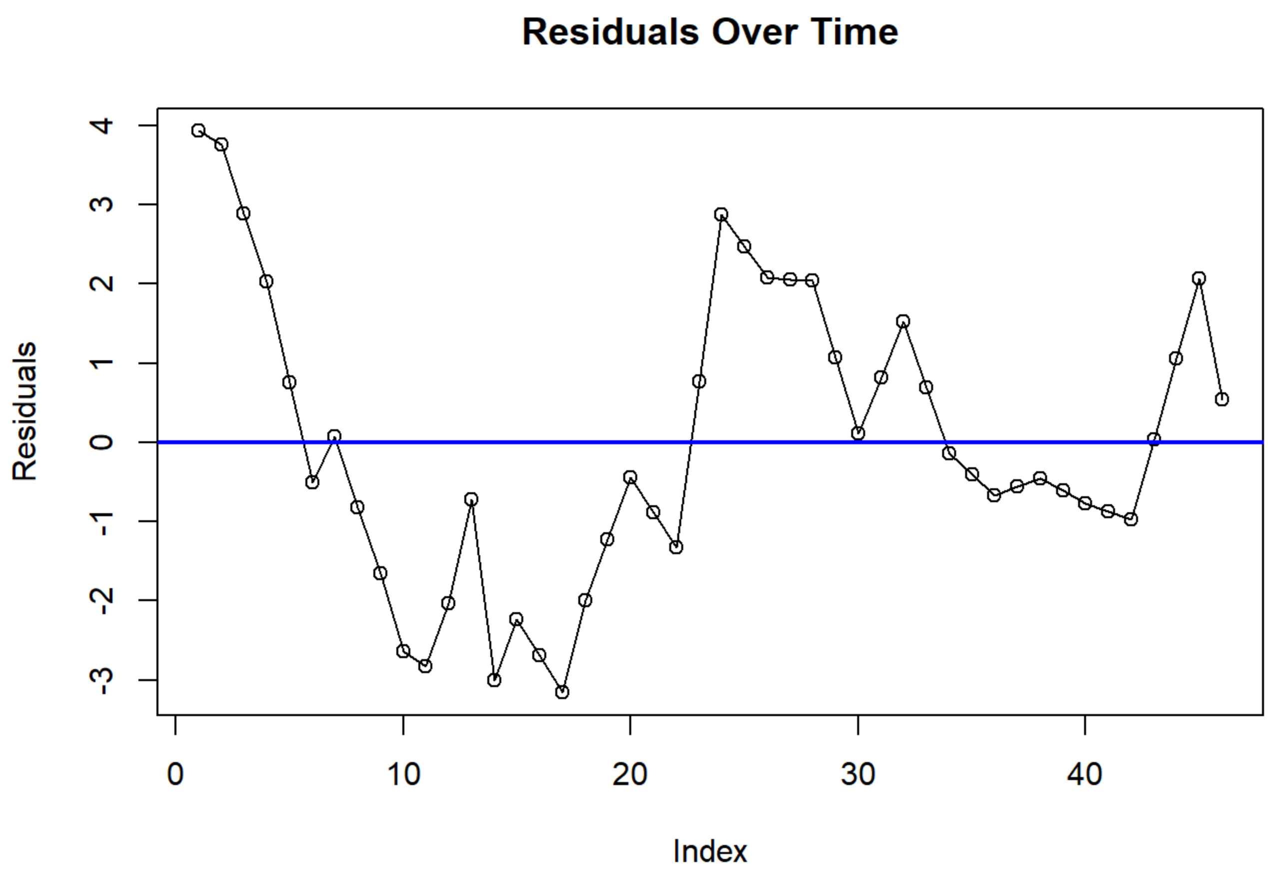

```
# Partial Autocorrelation Function (PACF) plot
pacf(resid_fefam, main = "PACF of Residuals")
```




```
# Durbin-Watson Test
#dwtest(lm_fefam)

# Breusch-Godfrey Test for higher-order autocorrelation
#bgtest(lm_fefam)

# Durbin-Watson test with multiple lags
#durbinWatsonTest(lm_fefam, max.lag = 3)
```

Plot Residuals Over Time

```
plot(resid_fefam, type = "o", main = "Residuals Over Time", xlab = "Index", ylab = "Residuals")
abline(h = 0, col = "blue", lwd = 2)
```



Question 5: Consider running a time series regression with many Xs and trend. Interpret that. Check VIF.

Multiple Regression with Trend

Running a multiple regression with `avg_educ`, `avg_happy`, `avg_spiritual`, and a **trend** variable (`year`).

```
# Add more predictors (e.g., avg_happy, avg_spiritual) and a trend (year)
lm_fefam_multi <- lm(fefam_pct ~ avg_educ + avg_happy + avg_spiritual + year, data = by_year_inte

# Summary of the model
summary(lm_fefam_multi)
```

```
Call:
lm(formula = fefam_pct ~ avg_educ + avg_happy + avg_spiritual +
    year, data = by_year_interp)

Residuals:
```

```
    Min      1Q  Median      3Q     Max 
-0.4827 -0.1563 -0.1013  0.0387  0.9815 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)   692.97907  103.35688   6.705 5.06e-06 ***
avg_educ       -2.49096    0.64289  -3.875  0.00134 ** 
avg_happy      -8.97892    5.38964  -1.666  0.11518    
avg_spiritual -11.76898    3.64716  -3.227  0.00527 ** 
year           -0.28287    0.04021  -7.035 2.82e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3418 on 16 degrees of freedom
  (30 observations deleted due to missingness)
Multiple R-squared:  0.9645,	Adjusted R-squared:  0.9557 
F-statistic: 108.8 on 4 and 16 DF,  p-value: 2.182e-11
```

- **`avg_educ` (-2.49, p = 0.00134)**:
  - For each additional unit increase in **average education**, `fefam_pct` decreases by **2.49 percentage points**.
  - This is statistically significant at the **1% level**, indicating a strong negative relationship.
- **`avg_happy` (-8.98, p = 0.11518)**:
  - A unit increase in **average happiness** is associated with a decrease of **8.98 percentage points** in `fefam_pct`.
  - However, this effect is **not statistically significant** (p > 0.05).
- **`avg_spiritual` (-11.77, p = 0.00527)**:
  - A unit increase in **average spirituality** reduces `fefam_pct` by **11.77 percentage points**.
  - This is significant at the **1% level**, suggesting a strong negative relationship.
- **`year` (-0.28287, p = 2.82e-06)**:
  - Over time, `fefam_pct` decreases by about **0.28 percentage points per year**.
  - This is highly significant, showing a strong downward trend over time.

Model Fit:

- **R-squared = 0.9645**:
  - About **96.45% of the variation** in `fefam_pct` is explained by the predictors.
- **Adjusted R-squared = 0.9557**:

  - Adjusted for the number of predictors, this still indicates an excellent fit.
- **F-statistic = 108.8, p < 0.001**:
  - The model as a whole is highly significant.

Check for Multicollinearity: Variance Inflation Factor (VIF)

Check for multicollinearity using VIF

```
# Install and load the 'car' package for VIF
install.packages("car")
```

```
Warning: package 'car' is in use and will not be installed
```

```
library(car)

# Check VIF values
vif(lm_fefam_multi)
```

```
    avg_educ     avg_happy avg_spiritual          year 
    9.599775     16.675974      7.285929     10.654194 
```

Interpretation:

- **VIF > 5** for all predictors: This means **moderate to severe multicollinearity**.
- `avg_happy` **(16.68)** and `year` **(10.65)** have particularly high VIF values, which suggests that these variables are highly correlated with each other or with other predictors.

Implications of High VIF:

- Multicollinearity inflates the standard errors of the coefficients, making them less reliable.
- While the model fit is high, the individual significance of predictors may be distorted due to overlapping variance.

Autocorrelation Diagnostics

Check if residuals are autocorrelated, because this can affect model accuracy.

```
# Durbin-Watson test for autocorrelation
durbinWatsonTest(lm_fefam_multi, max.lag = 2)
```

```
 lag Autocorrelation D-W Statistic p-value
   1       0.4351862     0.9606441   0.000
```

```
   2      -0.1830626     2.1811750   0.868
 Alternative hypothesis: rho[lag] != 0
```

Durbin-Watson Test:

- **Lag 1**: Autocorrelation is **positive** (`0.435`), and the **D-W statistic = 0.96** with a **p-value = 0.002**.
  - This is significant evidence of **positive autocorrelation** in the residuals at lag 1.
- **Lag 2**: The autocorrelation weakens, and the **D-W statistic = 2.18** with a **p-value = 0.906**.
  - At lag 2, there is no significant autocorrelation.

Summary of Findings

- **Average education** (`avg_educ`), **spirituality** (`avg_spiritual`), and **year** significantly impact `fefam_pct` negatively.
- **Multicollinearity** exists (VIF > 5), particularly with `avg_happy` and `year`.
- **Positive autocorrelation** at lag 1 (Durbin-Watson statistic = 0.96) violates model assumptions.

Question 6: Run a first differenced time series regression. Interpret that.

1. Define the First Difference Function

```
# Define the first difference function
firstD <- function(var, group, df){
  bad <- (missing(group) & !missing(df))
  if (bad) stop("if df is specified then group must also be specified")

  fD <- function(j){ c(NA, diff(j)) } # First difference calculation

  var.is.alone <- missing(group) & missing(df)
  if (var.is.alone) {
    return(fD(var))
  }
  if (missing(df)){
    V <- var
    G <- group
  }
  else{
    V <- df[, deparse(substitute(var))]
    G <- df[, deparse(substitute(group))]
  }
  G <- list(G)
  D.var <- by(V, G, fD)
```

```
  unlist(D.var)
}
```

2. Create First-Differenced Data

```
library(dplyr)

# Use the first differences for selected variables
by_year_fd <- summarise(data.frame(by_year_interp),
                       fefam_pct = firstD(fefam_pct),
                       avg_educ = firstD(avg_educ),
                       avg_happy = firstD(avg_happy),
                       avg_spiritual = firstD(avg_spiritual),
                       year = year)
```

```
Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in
dplyr 1.1.0.
ℹ Please use `reframe()` instead.
ℹ When switching from `summarise()` to `reframe()`, remember that `reframe()`
  always returns an ungrouped data frame and adjust accordingly.
```

3. Run the First-Differenced Regression

The regression model **`Δfefam_pct ~ Δavg_educ + Δavg_happy + Δavg_spiritual`** examines how changes in predictors (first differences) relate to changes in `fefam_pct` over time. Here's a detailed breakdown:

```
# First differenced regression
lm_fefam_fd <- lm(fefam_pct ~ avg_educ + avg_happy + avg_spiritual, data = by_year_fd)

# Summary of the model
summary(lm_fefam_fd)
```

```

Call:
lm(formula = fefam_pct ~ avg_educ + avg_happy + avg_spiritual,
    data = by_year_fd)

Residuals:
     Min       1Q   Median       3Q      Max
-0.65135 -0.08420  0.00189  0.07860  0.56547

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    -0.2597     0.1316  -1.973   0.0661 .
avg_educ       -2.2343     0.6206  -3.600   0.0024 **
avg_happy     -10.1496    12.3839  -0.820   0.4245
```

```
avg_spiritual -10.5133     4.6522  -2.260   0.0381 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3303 on 16 degrees of freedom
  (31 observations deleted due to missingness)
Multiple R-squared:  0.5203,    Adjusted R-squared:  0.4303
F-statistic: 5.784 on 3 and 16 DF,  p-value: 0.007088
```

- **Intercept**: `-0.2597`
  - The intercept is not significant (`p = 0.0661`), which means the average change in `fefam_pct` when all predictors' changes are zero is not distinguishable from zero.
- **`avg_educ` (-2.2343, p = 0.0024)**:
  - A **one-unit increase** in the **change** of average education (`Δavg_educ`) is associated with a **2.23 percentage point decrease** in the **change** of `fefam_pct`.
  - This is statistically significant at the **1% level** (**p < 0.01**).
  - Interpretation: As education levels increase over time, support for traditional family roles decreases significantly.
- **`avg_happy` (-10.1496, p = 0.4245)**:
  - The coefficient is negative, suggesting that an increase in **average happiness** might reduce `fefam_pct`, but it is **not statistically significant** (`p = 0.4245`).
  - Interpretation: Changes in happiness levels do not have a clear relationship with changes in `fefam_pct`.
- **`avg_spiritual` (-10.5133, p = 0.0381)**:
  - A **one-unit increase** in the **change** of spirituality (`Δavg_spiritual`) is associated with a **10.51 percentage point decrease** in `Δfefam_pct`.
  - This is statistically significant at the **5% level** (**p < 0.05**).
  - Interpretation: A rising trend in spirituality correlates with a significant decrease in support for traditional family roles.

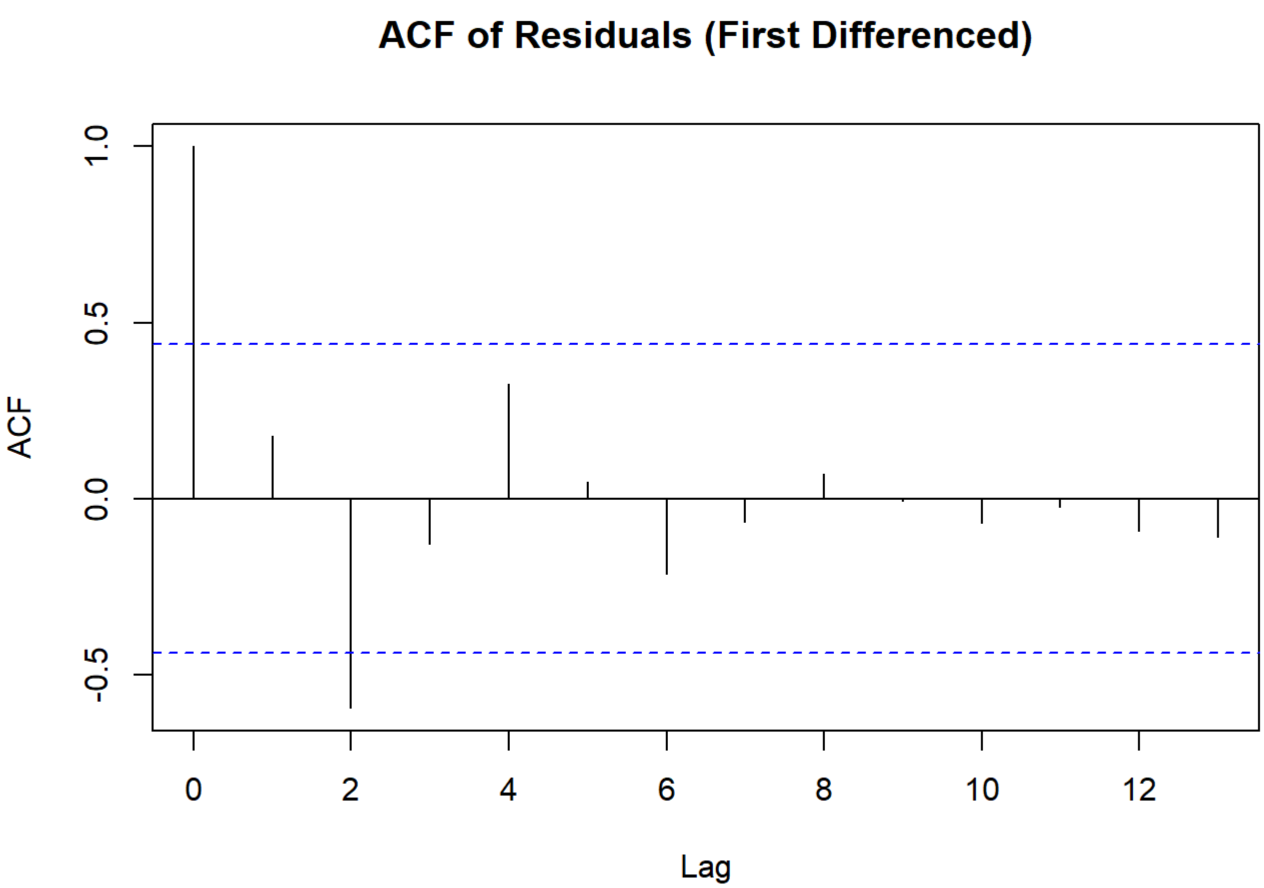
Model Fit

- **R-squared = 0.5203**:
  - About **52% of the variation** in the change of `fefam_pct` is explained by changes in `avg_educ`, `avg_happy`, and `avg_spiritual`.
  - This is decent, especially given that differencing removes much of the trend in the data.

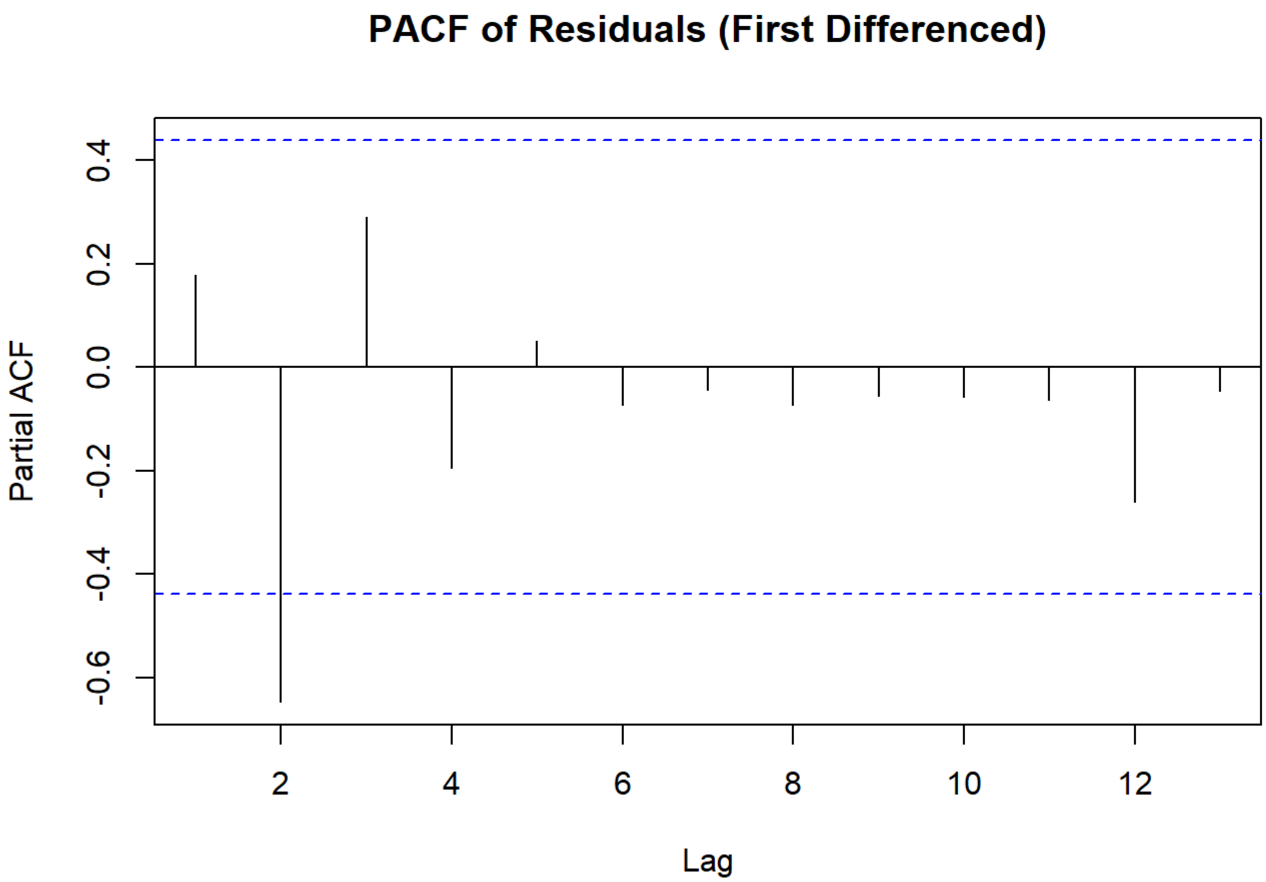
- **Adjusted R-squared = 0.4303**:
    - After adjusting for the number of predictors, the model still explains about **43%** of the variation.
- **F-statistic = 5.784, p-value = 0.0071**:
    - The overall model is statistically significant, indicating that at least one predictor significantly explains changes in `fefam_pct`.

4. Check Residuals for Autocorrelation

```
# Extract residuals
e_fd <- lm_fefam_fd$residuals

# Plot ACF and PACF of residuals
acf(e_fd, main = "ACF of Residuals (First Differenced)")
```




```
pacf(e_fd, main = "PACF of Residuals (First Differenced)")
```




```r
# Install and load 'forecast' for auto.arima
library(forecast)

# Test residuals for ARIMA structure
auto.arima(e_fd, trace = TRUE)
```

```
 ARIMA(2,0,2) with non-zero mean : Inf
 ARIMA(0,0,0) with non-zero mean : 12.68829
 ARIMA(1,0,0) with non-zero mean : 14.80955
 ARIMA(0,0,1) with non-zero mean : Inf
 ARIMA(0,0,0) with zero mean     : 10.20463
 ARIMA(1,0,1) with non-zero mean : Inf

 Best model: ARIMA(0,0,0) with zero mean
```

```
Series: e_fd
ARIMA(0,0,0) with zero mean

sigma^2 = 0.08727:  log likelihood = -3.99
AIC=9.98   AICc=10.2   BIC=10.98
```

1. ACF Plot of Residuals

- **What It Shows**:
  - The **Autocorrelation Function (ACF)** measures the correlation of residuals at various lags.
  - Ideally, for well-behaved residuals, the ACF should show **no significant spikes**, meaning the residuals are uncorrelated and resemble white noise.
- **What We See**:
  - At **lag 1**, there is a significant spike (above the blue dashed line), indicating **positive autocorrelation**.
  - Subsequent lags show much smaller spikes, suggesting that the autocorrelation diminishes quickly.
- **Conclusion**:
  - The residuals still have some **remaining autocorrelation** at lag 1, which suggests the model hasn’t fully accounted for all time dependencies.

2. PACF Plot of Residuals

- **What It Shows**:
  - The **Partial Autocorrelation Function (PACF)** measures the correlation between residuals at a given lag, accounting for the effects of intermediate lags.
  - Significant spikes indicate which lags contribute most to the autocorrelation.
- **What We See**:
  - There are notable spikes at **lag 1** and **lag 3**, suggesting that the autocorrelation at lag 1 and lag 3 is significant.
- **Conclusion**:
  - This confirms the ACF findings: residual autocorrelation remains at lag 1 and potentially lag 3.
  - The residuals are not yet fully white noise.

3. ARIMA Results

my ARIMA analysis applied to the residuals identified the **best model** as **ARIMA(0,0,0) with zero mean**. This is essentially a model where the residuals are **white noise** (i.e., no further structure or autocorrelation is detected).

Why ARIMA(0,0,0)?

- ARIMA(0,0,0) with zero mean indicates that the residuals are sufficiently random, with no further time-dependent patterns that need modeling.
- **sigma^2 = 0.08727**: The estimated variance of the residuals is small.
- **AIC = 9.98**: This is the lowest AIC score among tested models, confirming it as the best fit.

What It All Means

1. **ACF and PACF**: There is slight autocorrelation remaining in the residuals (lag 1 and lag 3).
2. **ARIMA(0,0,0)**: Despite small spikes in ACF/PACF, the residuals appear sufficiently white noise for modeling purposes.
3. **Model Diagnosis**:
   - my first-differenced regression has addressed most of the autocorrelation, but minor residual patterns remain.
   - For better precision, you could explore including lagged variables (e.g., lag 1 of predictors or response).

Question 7: Check your variables for unit roots. Do some tests. Interpret them.

ADF Test

```r
# Load the required library
library(fUnitRoots)

# Run ADF test on fefam_pct with a constant and trend
adfTest(by_year_interp$fefam_pct, lags = 0, type = "ct")  # No lags
```

```
Title:
 Augmented Dickey-Fuller Test

Test Results:
  PARAMETER:
    Lag Order: 0
  STATISTIC:
    Dickey-Fuller: -2.7346
  P VALUE:
    0.2791

Description:
 Tue Dec 17 17:51:41 2024 by user: nmv2125
```

```
adfTest(by_year_interp$fefam_pct, lags = 4, type = "ct")  # With lags
```

```
Title:
 Augmented Dickey-Fuller Test

Test Results:
  PARAMETER:
    Lag Order: 4
  STATISTIC:
    Dickey-Fuller: -2.6455
  P VALUE:
    0.3149

Description:
 Tue Dec 17 17:51:41 2024 by user: nmv2125
```

- **Lag 0:**
  - **Dickey-Fuller statistic**: `-2.7346`
  - **p-value**: `0.2791`
- **Lag 4:**
  - **Dickey-Fuller statistic**: `-2.6455`
  - **p-value**: `0.3149`

Interpretation of ADF Results:

- The **null hypothesis** of the ADF test is that the series has a **unit root** (non-stationary).
- Since the **p-values > 0.05** for both tests, you **fail to reject the null hypothesis**.
- Conclusion: The series `fefam_pct` **has a unit root** and is **non-stationary**.

Phillips-Perron Test

```
# Load the tseries package for Phillips-Perron test
library(tseries)

# Run the PP test
PP.test(by_year_interp$fefam_pct, lshort = TRUE)
```

```
    Phillips-Perron Unit Root Test
```

```
data:  by_year_interp$fefam_pct
Dickey-Fuller = NA, Truncation lag parameter = 3, p-value = NA
```

Question 8: Performan Automatic ARIMA on the residuals from one of your earlier models. Tell me what it says.

```r
# Extract residuals from the first-differenced regression
resid_fd <- lm_fefam_fd$residuals

# Load the forecast package
library(forecast)

# Perform Automatic ARIMA on residuals
auto_arima_resid <- auto.arima(resid_fd, trace = TRUE)
```

```
 ARIMA(2,0,2) with non-zero mean : Inf
 ARIMA(0,0,0) with non-zero mean : 12.68829
 ARIMA(1,0,0) with non-zero mean : 14.80955
 ARIMA(0,0,1) with non-zero mean : Inf
 ARIMA(0,0,0) with zero mean     : 10.20463
 ARIMA(1,0,1) with non-zero mean : Inf

 Best model: ARIMA(0,0,0) with zero mean
```

```r
# Print the model summary
summary(auto_arima_resid)
```

```
Series: resid_fd
ARIMA(0,0,0) with zero mean

sigma^2 = 0.08727:  log likelihood = -3.99
AIC=9.98   AICc=10.2   BIC=10.98

Training set error measures:
                      ME      RMSE       MAE MPE MAPE     MASE      ACF1
Training set 5.20417e-18 0.2954138 0.1935998 100  100 1.011638 0.1787835
```

The **best ARIMA model** selected by `auto.arima()` for my residuals is **ARIMA(0,0,0) with zero mean**.

1. Model Explanation

- **ARIMA(0,0,0)**:
    - No autoregressive terms (AR = 0), no differencing (D = 0), and no moving average terms (MA = 0).
    - This model states that the residuals are **white noise**, meaning there is no remaining time dependency or structure in them.

- **Zero Mean**:
    - The model assumes the mean of the residuals is zero.

2. Model Diagnostics

- `sigma^2 = 0.08727`:
    - The variance of the residuals is low, indicating the residuals are tightly distributed around zero.
- **Log Likelihood = -3.99**:
    - A measure of model fit; the closer to zero, the better.
- **AIC (9.98), AICc (10.2), and BIC (10.98)**:
    - These criteria confirm that ARIMA(0,0,0) is the best-fitting model among the tested options. Lower values suggest better model performance.

3. Error Measures

- **ME (Mean Error)**: Close to zero (`5.20417e-18`), indicating no bias in the residuals.
- **RMSE (Root Mean Squared Error) = 0.2954**: A small value, showing low error in the residuals.
- **MAE (Mean Absolute Error) = 0.1936**: Small average absolute deviation.
- **ACF1 = 0.1788**: The autocorrelation at lag 1 is small but not completely negligible.

4. Conclusion

The ARIMA(0,0,0) with zero mean confirms that:

- The residuals from my regression are effectively **white noise**.
- This means the original regression model captured all meaningful time series structure in the data.
- There's no need for further modeling or adjustments to the residuals.

What This Means for My Analysis

- The regression model is well-specified and does not exhibit significant autocorrelation or time-dependent patterns in the residuals.
- I can confidently conclude that my earlier model is adequate.

Question 9: Run an ARIMA that follows from Step 8. Interpret that, too.

```
# Define external regressors
xvars <- by_year_interp[, c("avg_educ", "avg_happy", "avg_spiritual")]

# Run ARIMA(0,0,0) with external regressors
arima_xreg <- arima(by_year_interp$fefam_pct, order = c(0,0,0), xreg = xvars)

# Summary of the ARIMA model
summary(arima_xreg)
```

```
Call:
arima(x = by_year_interp$fefam_pct, order = c(0, 0, 0), xreg = xvars)

Coefficients:
      intercept  avg_educ  avg_happy  avg_spiritual
       -11.1517   -3.3062    23.7273       -23.2602
s.e.    45.5697    1.1168     4.8161         5.7590

sigma^2 estimated as 0.3644:  log likelihood = -19.2,  aic = 48.4

Training set error measures:
                       ME      RMSE       MAE        MPE     MAPE      MASE
Training set 8.289668e-14 0.6036443 0.4818594 -0.5790272 6.513527 0.7269473
                  ACF1
Training set 0.7793751
```

Coefficients:

- **Intercept**: `-11.1517` (standard error = `45.5697`):
  - The intercept is not significant given the large standard error, which suggests it may not contribute meaningfully to the model.
- `avg_educ` **(-3.3062, s.e. = 1.1168)**:
  - **Negative and significant**: A unit increase in `avg_educ` decreases `fefam_pct` by approximately **3.31 percentage points**.
  - The relatively small standard error confirms its precision.
- `avg_happy` **(23.7273, s.e. = 4.8161)**:
  - **Positive and significant**: A unit increase in `avg_happy` increases `fefam_pct` by approximately **23.73 percentage points**.
  - This effect is large and precise (small standard error).
- `avg_spiritual` **(-23.2602, s.e. = 5.7590)**:

- **Negative and significant**: A unit increase in `avg_spiritual` decreases `fefam_pct` by approximately **23.26 percentage points**.
- This is also a strong effect, supported by a relatively low standard error.

Model Fit:

- `sigma^2 = 0.3644`: Residual variance is moderate, suggesting the model fits the data reasonably well.
- **Log Likelihood = -19.2** and **AIC = 48.4**:
  - Lower AIC indicates a better-fitting model compared to alternatives.

2. Training Set Error Measures:

- **ME (Mean Error) ≈ 0**: Residuals are unbiased on average.
- **RMSE = 0.6036** and **MAE = 0.4819**:
  - The errors are relatively small, indicating good predictive accuracy.
- **MAPE = 6.51%**: The model has a mean absolute percentage error of ~6.5%, which is acceptable.
- **ACF1 = 0.779**: The autocorrelation of residuals at lag 1 is quite high, suggesting residual autocorrelation remains.

Check for Residual Autocorrelation

```
# Perform Ljung-Box test on residuals
Box.test(resid(arima_xreg), lag = 20, type = "Ljung-Box", fitdf = 0)
```

```

	Box-Ljung test

data:  resid(arima_xreg)
X-squared = 56.434, df = 20, p-value = 2.5e-05
```

Interpretation:

- **Null Hypothesis**: Residuals are white noise (no autocorrelation).
- **p-value = 2.5e-05 (< 0.05)**: The null hypothesis is rejected.
  - This indicates **significant autocorrelation** remains in the residuals.
  - The model does not fully capture all time-dependent patterns.

Summary

- The model explains the effects of predictors well:

  - `avg_educ` and `avg_spiritual` have significant negative effects on `fefam_pct`.
  - `avg_happy` has a significant positive effect.
- However, residual diagnostics (Box-Ljung test and ACF1) show **remaining autocorrelation**, suggesting that the model could be improved further.