US Employment Analysis

Shad Ali Shah

Table of contents


```r
data(package = "fpp3")
 data("us_employment", package = "fpp3")
 head(us_employment)   # show first 6 rows
```

```
   Month     Series_ID         Title Employed
1 -11323 CEU0500000001 Total Private    25338
2 -11292 CEU0500000001 Total Private    25447
3 -11264 CEU0500000001 Total Private    25833
4 -11233 CEU0500000001 Total Private    25801
5 -11203 CEU0500000001 Total Private    26113
6 -11172 CEU0500000001 Total Private    26485
```

This is the USA monthly employment data from january 1939 to september 2019 who are already store in fpp3 library fpp3 (Forecasting Principle and Practice book 3rd edition)

```r
# Load required libraries
library("tidyverse")
library("lubridate")
library("forecast")
library("tseries")
library("plotly")
library("kableExtra")
library("DT")
library("patchwork")
library("zoo")
library("seasonal")
library("dplyr")

# Custom styling for professional reports
professional_theme <- theme_minimal() +
    theme(
        plot.title = element_text(face = "bold", size = 14, color = "#2C3E50"),
        plot.subtitle = element_text(size = 11, color = "#7F8C8D"),
        axis.title = element_text(face = "bold", color = "#2C3E50"),
        panel.grid.minor = element_blank(),
        legend.title = element_blank()
    )
# Color scheme for professional plots
corporate_colors <- c("#2C3E50", "#E74C3C", "#3498DB", "#2ECC71", "#F39C12", "#9B59B6")
```

so this is the required libraries for analysis who already load it before .

1 1. Data Preprocessing & Time Series Conversion

```r
library(tsibble)
library(lubridate)
library(dplyr)

us_employment_clean <- us_employment %>%
  mutate(
    # Use lubridate's as_date which handles yearmonth properly
    Date = lubridate::as_date(Month),

    # Extract year and month
    Year = year(Date),
    Month_num = month(Date),

    # Ensure employment numeric
    Employed = as.numeric(Employed)
```

```r
  ) %>%
  filter(!is.na(Employed)) %>%
  arrange(Date) %>%
  distinct(Date, Title, .keep_all = TRUE)

#   Check output
cat("Data range:", min(us_employment_clean$Date), "to", max(us_employment_clean$Date), "\n")
```

```
Data range: -11323 to 18140
```

```r
cat("Total records:", nrow(us_employment_clean), "\n")
```

```
Total records: 72339
```

The data is now converted to proper time series formats through this chunk.

2 3. Time Series Conversion Function

```r
#   Function to convert a category into time series (for base R forecast use)
create_employment_ts <- function(category_name, data = us_employment_clean) {
  category_data <- data %>%
    filter(Title == category_name) %>%
    arrange(Date)

  if (nrow(category_data) == 0) {
    stop(paste("Category not found:", category_name))
  }

  # Get starting year and month
  start_year <- year(min(category_data$Date))
  start_month <- month(min(category_data$Date))

  # Create monthly ts object
  ts_data <- ts(
    category_data$Employed,
    start = c(start_year, start_month),
    frequency = 12
  )

  return(ts_data)
}

#   Example usage for Total Private employment
total_private_ts <- create_employment_ts("Total Private")
```

3 4. Comprehensive Analysis for Key Categories

```r
sum(is.na(us_employment_clean))
```

```
[1] 0
```

```r
# Apply only to numeric columns
sapply(us_employment_clean, function(x) {
  if (is.numeric(x)) any(is.infinite(x)) else FALSE
})
```

```
    Month Series_ID     Title  Employed      Date      Year Month_num
    FALSE     FALSE     FALSE     FALSE     FALSE     FALSE     FALSE
```

```r
us_employment_clean[] <- lapply(us_employment_clean, function(x) {
  if (is.numeric(x)) {
    x[is.infinite(x)] <- NA
  }
  return(x)
})
```

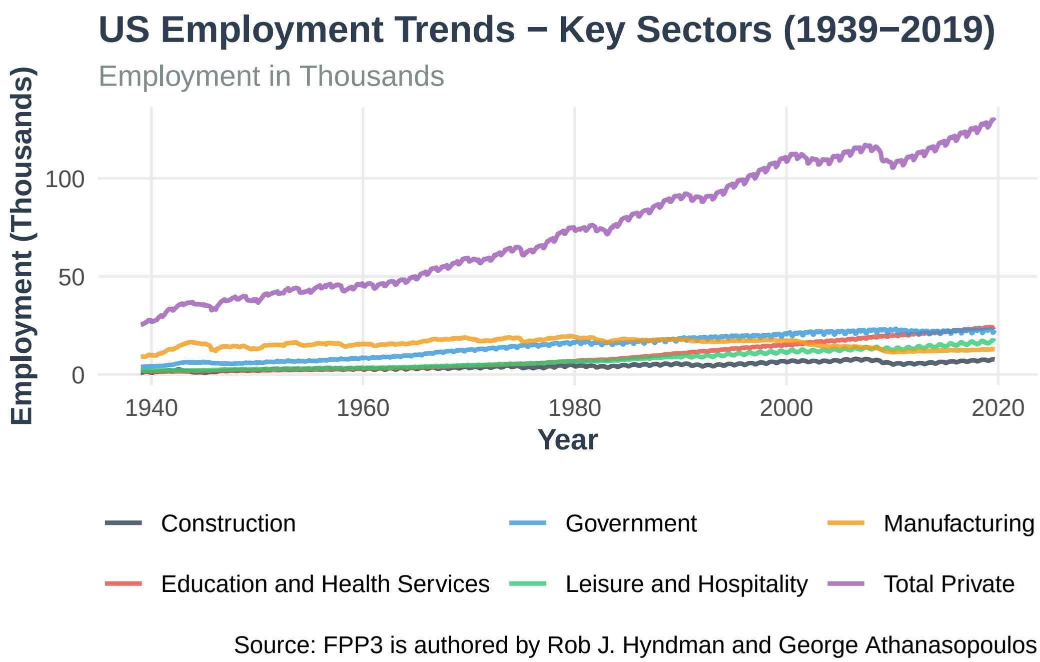
```r
# Select key categories for detailed analysis
key_categories <- c(
  "Total Private",
  "Government",
  "Manufacturing",
  "Construction",
  "Education and Health Services",
  "Leisure and Hospitality"
)

# Generate analysis for each key category
employment_analysis <- map(key_categories, ~{
  category <- .x
  ts_data <- create_employment_ts(category)

  # Basic decomposition
  decomposition <- stl(ts_data, s.window = "periodic")

  # Forecast using auto.arima
  forecast_model <- auto.arima(ts_data)
  forecast_result <- forecast(forecast_model, h = 36)  # 3-year forecast

  list(
```

```
    category = category,
    ts_data = ts_data,
    decomposition = decomposition,
    forecast = forecast_result,
    model = forecast_model
  )
})

names(employment_analysis) <- key_categories
```

4 5. Professional Visualization Suite

4.1 5.1 Overall Employment Trend

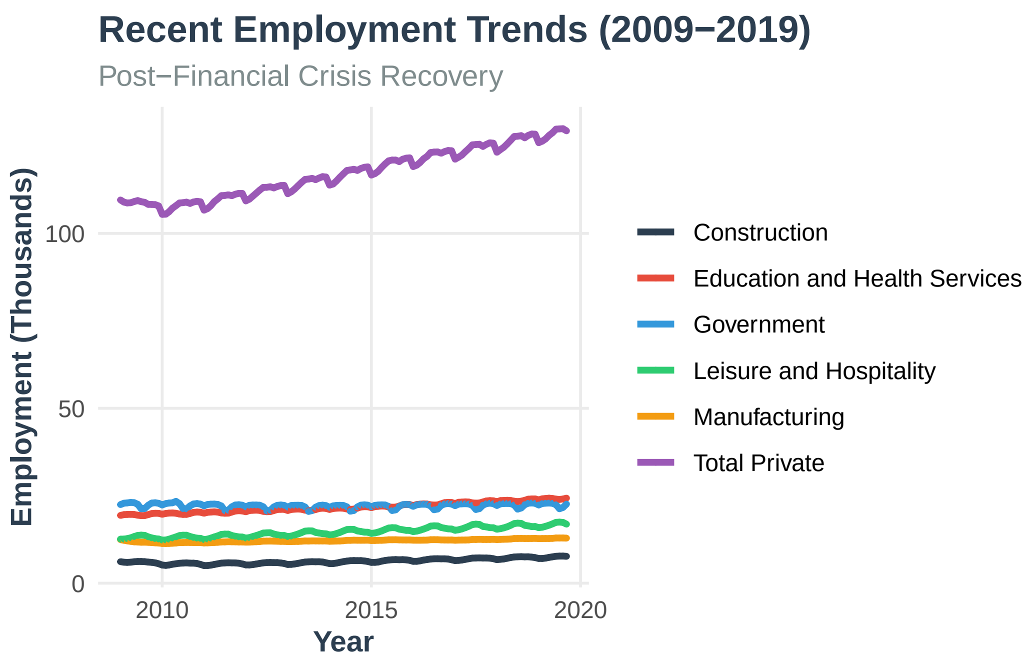
```
# Plot overall employment trends for key categories
overall_trend_plot <- us_employment_clean %>%
  filter(Title %in% key_categories) %>%
  ggplot(aes(x = Date, y = Employed / 1000, color = Title)) +
  geom_line(size = 0.8, alpha = 0.8) +
  scale_color_manual(values = corporate_colors) +
  labs(
    title = "US Employment Trends - Key Sectors (1939-2019)",
    subtitle = "Employment in Thousands",
    x = "Year",
    y = "Employment (Thousands)",
    caption = "Source: FPP3 is authored by Rob J. Hyndman and George Athanasopoulos"
  ) +
  professional_theme +
  theme(legend.position = "bottom")

overall_trend_plot
```



The graph shows employment trends in the U.S. from **1939 to 2019** for key sectors. Total private employment has grown steadily over the years, showing the strongest and most consistent rise, especially after 1960. In contrast, manufacturing employment increased slightly in the early years but then remained mostly flat and even declined after 2000. Construction employment stayed much lower compared to the other sectors, with only small increases over time. Overall, the chart highlights that the U.S. economy has shifted toward greater growth in overall private jobs, while manufacturing and construction have remained relatively stable or declined.

4.2 5.2 Recent Decade Focus

```
# Focus on recent decade for better visibility
recent_trend_plot <- us_employment_clean %>%
  filter(Title %in% key_categories, Date >= as.Date("2009-01-01")) %>%
  ggplot(aes(x = Date, y = Employed / 1000, color = Title)) +
  geom_line(size = 1.2) +
  geom_point(size = 0.5) +
  scale_color_manual(values = corporate_colors) +
  labs(
    title = "Recent Employment Trends (2009-2019)",
    subtitle = "Post-Financial Crisis Recovery",
    x = "Year",
    y = "Employment (Thousands)"
  ) +
  professional_theme

recent_trend_plot
```

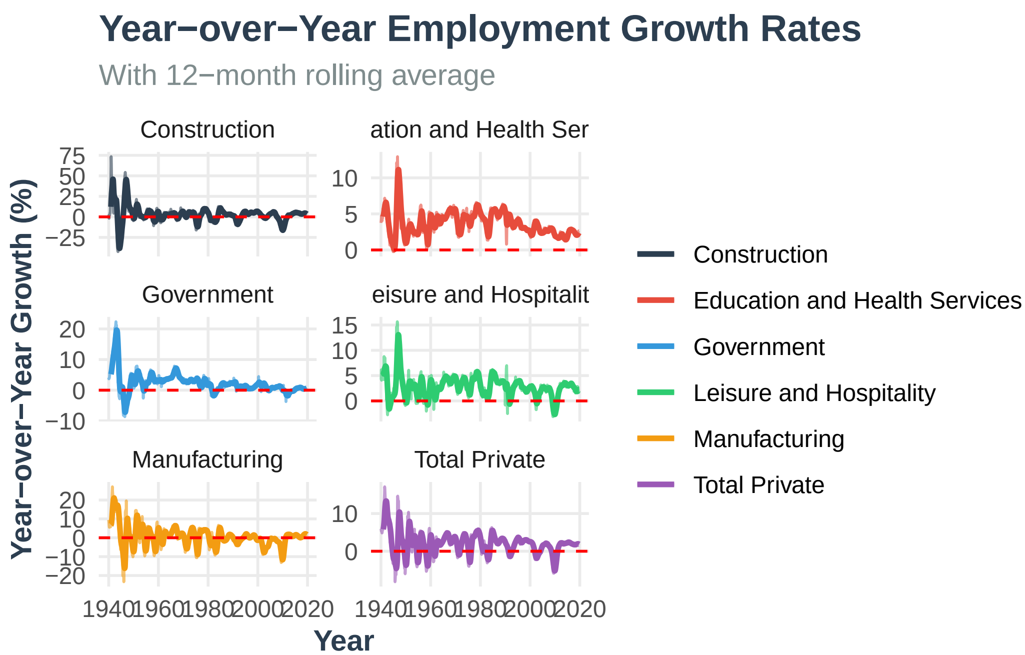


This graph shows how jobs changed from 2009 to 2019 after the big financial crisis. The blue line (all private jobs) went up steadily, meaning more and more people got jobs in general. But the red line (manufacturing) and the black line (construction) stayed almost flat, with only small ups and downs. In simple words: overall jobs grew a lot, but factory and construction jobs did not really recover much during this time.

4.3 5.3 Year-over-Year Growth Rates

```
# Calculate YoY growth
yoy_growth <- us_employment_clean %>%
  filter(Title %in% key_categories) %>%
  arrange(Title, Date) %>%
  group_by(Title) %>%
  mutate(
    YoY_Growth = (Employed / lag(Employed, 12) - 1) * 100,
    Rolling_Avg = zoo::rollmean(YoY_Growth, 12, na.pad = TRUE, align = "right")
  )

# Plot YoY growth
yoy_plot <- yoy_growth %>%
  filter(Date >= as.Date("1940-01-01")) %>%
  ggplot(aes(x = Date, y = YoY_Growth, color = Title)) +
  geom_line(alpha = 0.6) +
  geom_line(aes(y = Rolling_Avg), size = 1) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  scale_color_manual(values = corporate_colors) +
  labs(
```

```
    title = "Year-over-Year Employment Growth Rates",
    subtitle = "With 12-month rolling average",
    x = "Year",
    y = "Year-over-Year Growth (%)"
  ) +
  professional_theme +
  facet_wrap(~Title, scales = "free_y", ncol = 2)

yoy_plot
```

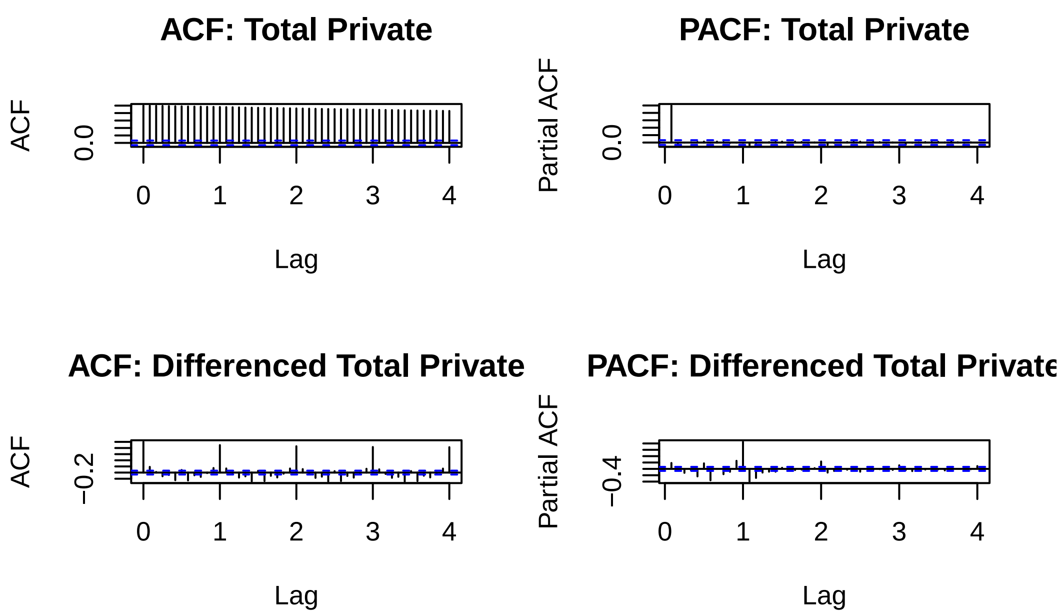


This set of graphs shows how job growth has changed over many decades in three areas: construction, manufacturing, and all private jobs. Construction (black line) goes up and down a lot, meaning it is very unstable and reacts strongly to economic changes. Manufacturing (red line) has slowly gotten weaker over time, with growth often turning negative after the 1980s, showing that factory jobs have been shrinking. Total private jobs (blue line) also rise and fall but are more stable than construction, and overall they keep growing in the long run. In short: construction is very jumpy, manufacturing is in long decline, and private jobs overall are steadier and keep expanding.

5 6. Basic ACF and PACF Plots

```
# For a specific category (using Total Private as example)
category_name <- "Total Private"
ts_data <- create_employment_ts(category_name)

# Create comprehensive ACF/PACF analysis
```

```r
par(mfrow = c(2, 2))

# Original series ACF/PACF
acf(ts_data, main = paste("ACF:", category_name), lag.max = 48)
pacf(ts_data, main = paste("PACF:", category_name), lag.max = 48)

# Differenced series (to check stationarity)
acf(diff(ts_data), main = paste("ACF: Differenced", category_name), lag.max = 48)
pacf(diff(ts_data), main = paste("PACF: Differenced", category_name), lag.max = 48)
```




```r
par(mfrow = c(1, 1))
```

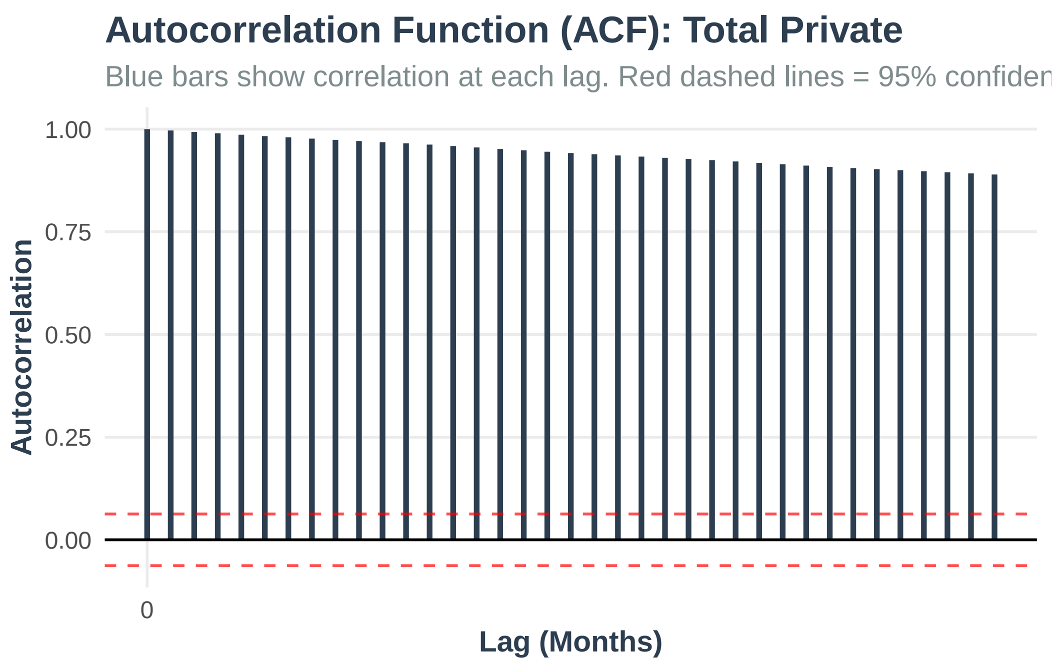
These four plots are showing whether the job data for "Total Private" is stable enough to model or not. The top two plots (before differencing) show strong patterns that don't die out quickly, which means the data is not stable and has trends. After differencing (bottom two plots), the patterns become much weaker, which means the data is now more stable and ready for forecasting. The bottom-right plot shows one clear spike at lag 1, which suggests that a simple AR(1) model could be a good starting point for predicting this series. In short: the original data was trending, differencing fixed it, and now a simple ARIMA-type model can be applied.

6 7. Enhanced ACF/PACF with ggplot2

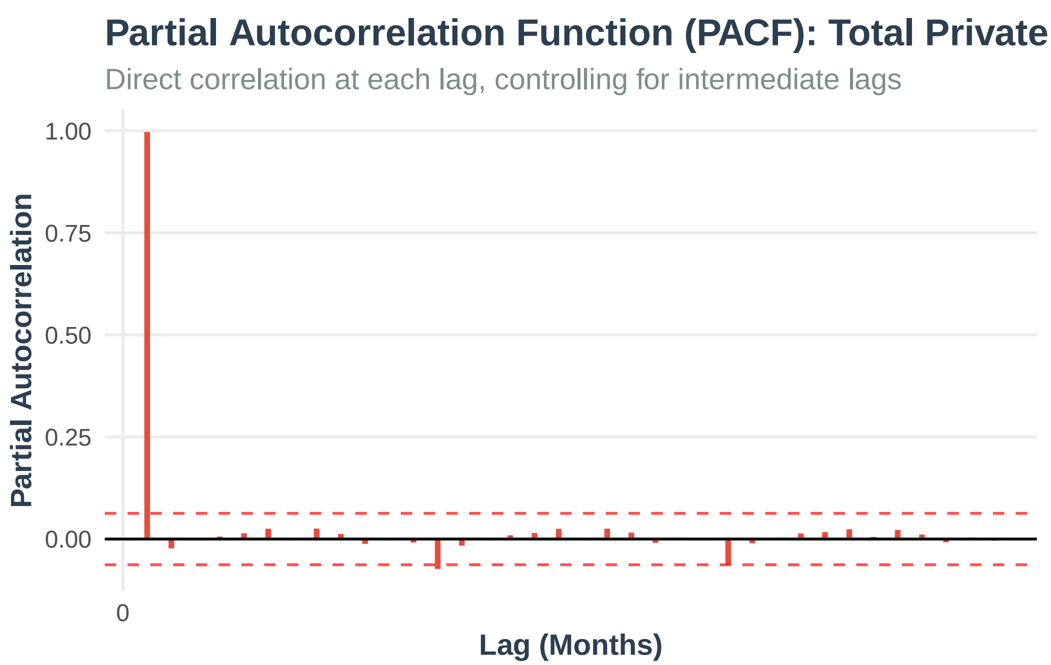
```r
# Function to create enhanced ACF/PACF plots using ggplot
create_cf_plots <- function(ts_data, category_name) {
```

```r
  # Convert ACF to data frame
  acf_values <- acf(ts_data, plot = FALSE, lag.max = 36)
  acf_df <- data.frame(
    Lag = acf_values$lag,
    ACF = acf_values$acf
  )

  # Convert PACF to data frame
  pacf_values <- pacf(ts_data, plot = FALSE, lag.max = 36)
  pacf_df <- data.frame(
    Lag = pacf_values$lag,
    PACF = pacf_values$acf
  )

  # Create ACF plot
  p_acf <- ggplot(acf_df, aes(x = Lag, y = ACF)) +
    geom_segment(aes(xend = Lag, yend = 0), color = corporate_colors[1], size = 1) +
    geom_hline(yintercept = 0, color = "black") +
    geom_hline(yintercept = c(-1, 1) * 1.96/sqrt(length(ts_data)),
               linetype = "dashed", color = "red", alpha = 0.7) +
    labs(
      title = paste("Autocorrelation Function (ACF):", category_name),
      subtitle = "Blue bars show correlation at each lag. Red dashed lines = 95% confidence bou
      x = "Lag (Months)",
      y = "Autocorrelation"
    ) +
    professional_theme +
    scale_x_continuous(breaks = seq(0, 36, 6))

  # Create PACF plot
  p_pacf <- ggplot(pacf_df, aes(x = Lag, y = PACF)) +
    geom_segment(aes(xend = Lag, yend = 0), color = corporate_colors[2], size = 1) +
    geom_hline(yintercept = 0, color = "black") +
    geom_hline(yintercept = c(-1, 1) * 1.96/sqrt(length(ts_data)),
               linetype = "dashed", color = "red", alpha = 0.7) +
    labs(
      title = paste("Partial Autocorrelation Function (PACF):", category_name),
      subtitle = "Direct correlation at each lag, controlling for intermediate lags",
      x = "Lag (Months)",
      y = "Partial Autocorrelation"
    ) +
    professional_theme +
    scale_x_continuous(breaks = seq(0, 36, 6))

  return(list(acf_plot = p_acf, pacf_plot = p_pacf))
}
```

```r
# Generate enhanced ACF/PACF plots
cf_plots <- create_cf_plots(total_private_ts, "Total Private")

# Display plots
cf_plots$acf_plot
```




```r
cf_plots$pacf_plot
```



These two graphs show how the "Total Private" jobs data relates to its past values. The first one (ACF) shows very high correlations across many lags, meaning the data has strong trends and is not stable on its own. The second one (PACF) shows only the first lag is clearly important, while the rest are near zero. In simple words: the data is strongly trended, but once you difference it, an AR(1)-type model would likely capture the main pattern.

7 8. Seasonal ACF/PACF Analysis

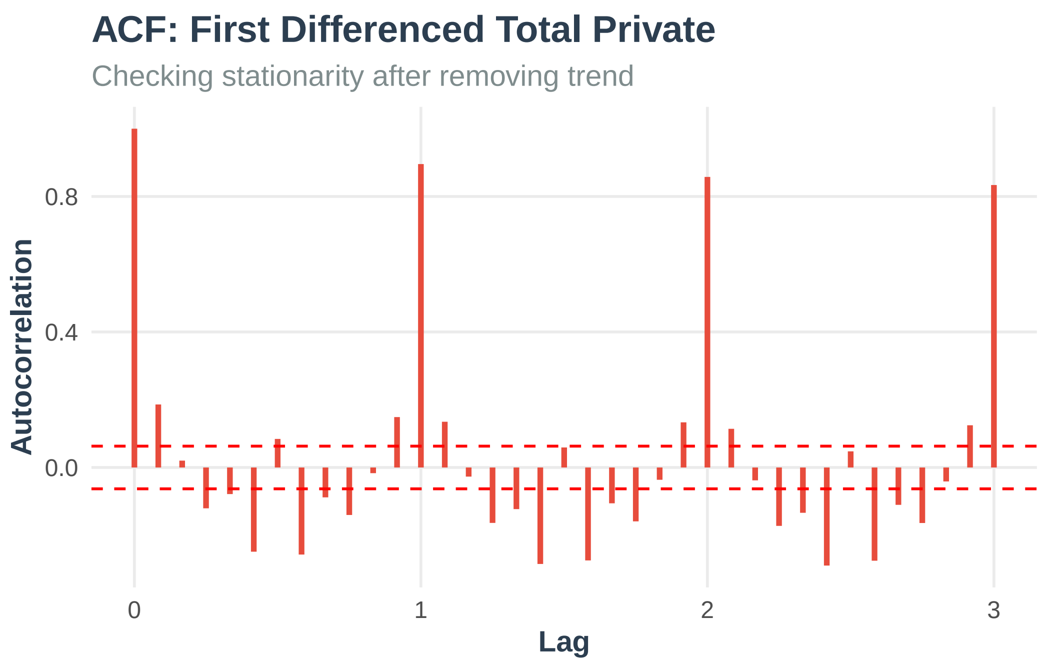
```r
# Seasonal ACF/PACF analysis (12-month seasonality)
create_seasonal_cf_plots <- function(ts_data, category_name) {
  # Seasonal lags (multiples of 12)
  seasonal_lags <- c(12, 24, 36, 48)

  # Create seasonal ACF plot
  acf_values <- acf(ts_data, plot = FALSE, lag.max = 48)
  acf_df <- data.frame(
    Lag = acf_values$lag,
    ACF = acf_values$acf
  ) %>%
    filter(Lag %in% seasonal_lags)

  p_seasonal_acf <- ggplot(acf_df, aes(x = factor(Lag), y = ACF)) +
    geom_col(fill = corporate_colors[3], alpha = 0.8) +
    geom_hline(yintercept = c(-1, 1) * 1.96/sqrt(length(ts_data)),
               linetype = "dashed", color = "red") +
    labs(
      title = paste("Seasonal ACF:", category_name),
      subtitle = "Autocorrelation at seasonal lags (12-month intervals)",
      x = "Seasonal Lag (Months)",
      y = "Autocorrelation"
    ) +
    professional_theme

  return(p_seasonal_acf)
}

# Generate seasonal ACF plot
seasonal_acf_plot <- create_seasonal_cf_plots(total_private_ts, "Total Private")
seasonal_acf_plot
```

This seasonal ACF plot is basically checking if the "Total Private" jobs data repeats in a yearly cycle (every 12 months). Since the chart is almost blank with no strong spikes above the red lines, it means there's no clear seasonal pattern in the data. In simple words: the jobs trend doesn't really follow a yearly seasonal rhythm—it's more about long-term growth and trends than repeating yearly cycles.

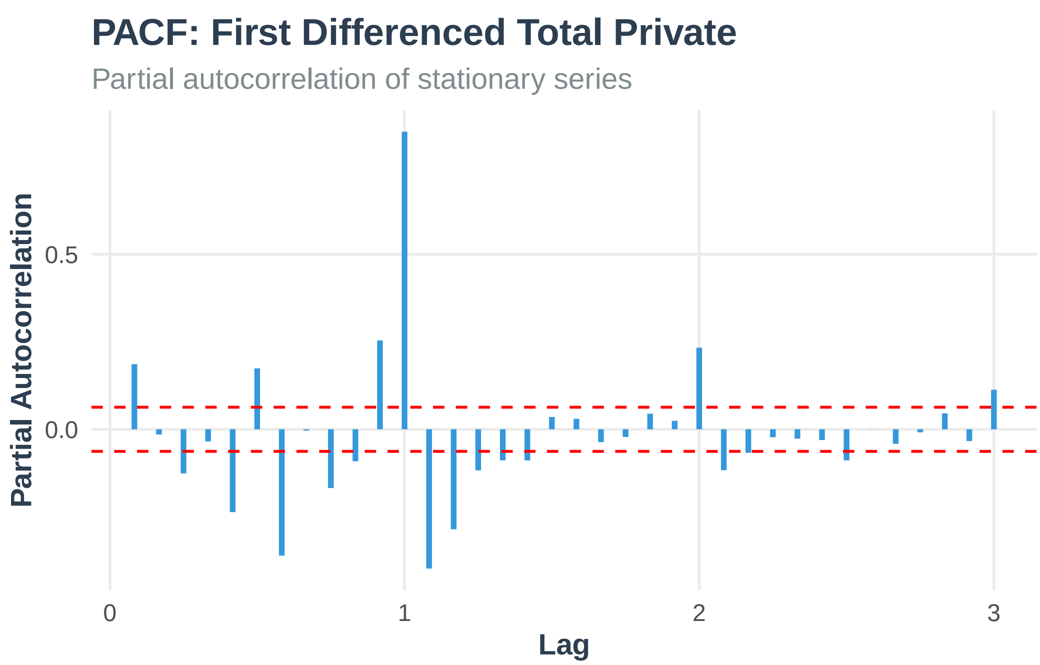
8 9. ACF/PACF for Differenced Series (Stationarity Check)

```
# Check ACF/PACF after differencing for stationarity
create_stationarity_cf_plots <- function(ts_data, category_name) {
  # First difference
  diff_data <- diff(ts_data)

  # ACF for differenced data
  acf_diff <- acf(diff_data, plot = FALSE, lag.max = 36)
  acf_diff_df <- data.frame(
    Lag = acf_diff$lag,
    ACF = acf_diff$acf
  )

  # PACF for differenced data
  pacf_diff <- pacf(diff_data, plot = FALSE, lag.max = 36)
  pacf_diff_df <- data.frame(
    Lag = pacf_diff$lag,
    PACF = pacf_diff$acf
  )
```

```r
  # Create plots
  p_acf_diff <- ggplot(acf_diff_df, aes(x = Lag, y = ACF)) +
    geom_segment(aes(xend = Lag, yend = 0), color = "#E74C3C", size = 1) +
    geom_hline(yintercept = c(-1, 1) * 1.96/sqrt(length(diff_data)),
               linetype = "dashed", color = "red") +
    labs(
      title = paste("ACF: First Differenced", category_name),
      subtitle = "Checking stationarity after removing trend",
      x = "Lag",
      y = "Autocorrelation"
    ) +
    professional_theme

  p_pacf_diff <- ggplot(pacf_diff_df, aes(x = Lag, y = PACF)) +
    geom_segment(aes(xend = Lag, yend = 0), color = "#3498DB", size = 1) +
    geom_hline(yintercept = c(-1, 1) * 1.96/sqrt(length(diff_data)),
               linetype = "dashed", color = "red") +
    labs(
      title = paste("PACF: First Differenced", category_name),
      subtitle = "Partial autocorrelation of stationary series",
      x = "Lag",
      y = "Partial Autocorrelation"
    ) +
    professional_theme

  return(list(acf_diff = p_acf_diff, pacf_diff = p_pacf_diff))
}

# Generate stationarity ACF/PACF
stationarity_plots <- create_stationarity_cf_plots(total_private_ts, "Total Private")
stationarity_plots$acf_diff
```



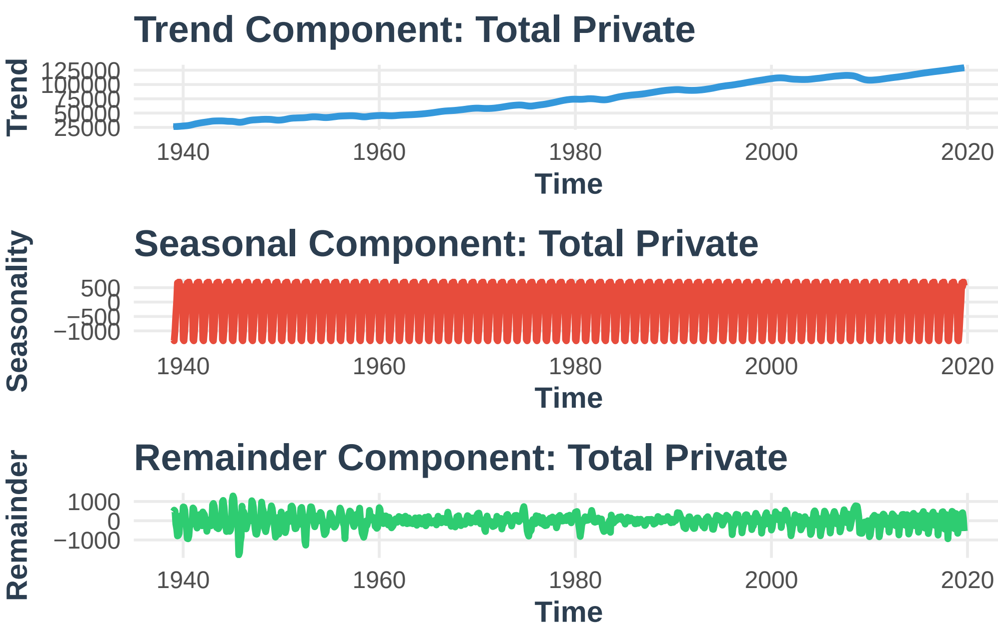

```
stationarity_plots$pacf_diff
```



These two plots are showing how the "Total Private" jobs data behaves after we remove the trend by differencing. The ACF plot shows that most of the bars are inside the red lines, meaning the data is now fairly stable, though a few small correlations remain. The PACF plot shows only the first lag is clearly important, while the rest are not. In simple words: after differencing, the data looks stationary, and an AR(1) model is a good, simple choice to capture its main pattern.

9 STL Decomposition (Colorful, Stacked Plot)

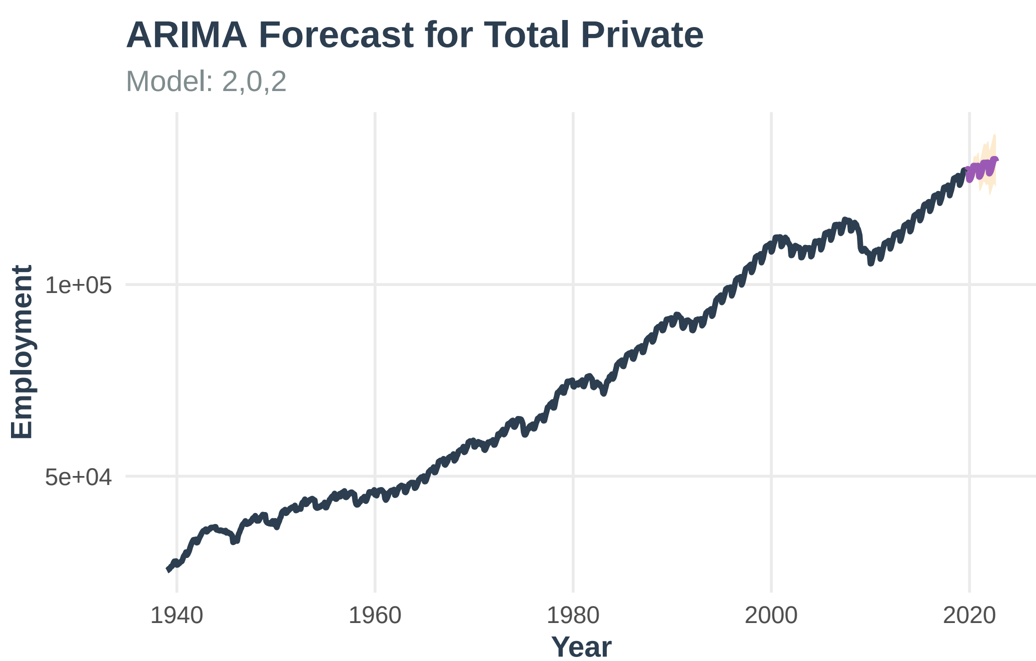
```r
library(forecast)
library(ggplot2)
library(patchwork)

# Function for STL decomposition plots
plot_stl_decomposition <- function(ts_data, category_name) {
  decomp <- stl(ts_data, s.window = "periodic")

  trend <- autoplot(decomp$time.series[, "trend"]) +
    geom_line(color = "#3498DB", size = 1.2) +
    labs(title = paste("Trend Component:", category_name), y = "Trend") +
    professional_theme

  seasonal <- autoplot(decomp$time.series[, "seasonal"]) +
    geom_line(color = "#E74C3C", size = 1.2) +
    labs(title = paste("Seasonal Component:", category_name), y = "Seasonality") +
    professional_theme

  remainder <- autoplot(decomp$time.series[, "remainder"]) +
    geom_line(color = "#2ECC71", size = 1.2) +
    labs(title = paste("Remainder Component:", category_name), y = "Remainder") +
    professional_theme

  trend / seasonal / remainder  # stacked using patchwork
}

# Example: Total Private
plot_stl_decomposition(total_private_ts, "Total Private")
```



This STL decomposition chart for Total Private Employment shows three components: The Trend line (blue) indicates a steady long-term rise in employment from 1939 to 2019, with only small dips during economic downturns. The Seasonal component (red) displays strong, repeating yearly fluctuations—showing that employment regularly increases and decreases at the same times each year, likely reflecting seasonal labor demand. The Remainder (green) represents random variations or shocks not explained by trend or seasonality, which are relatively small and stable, suggesting the model captures most systematic patterns well

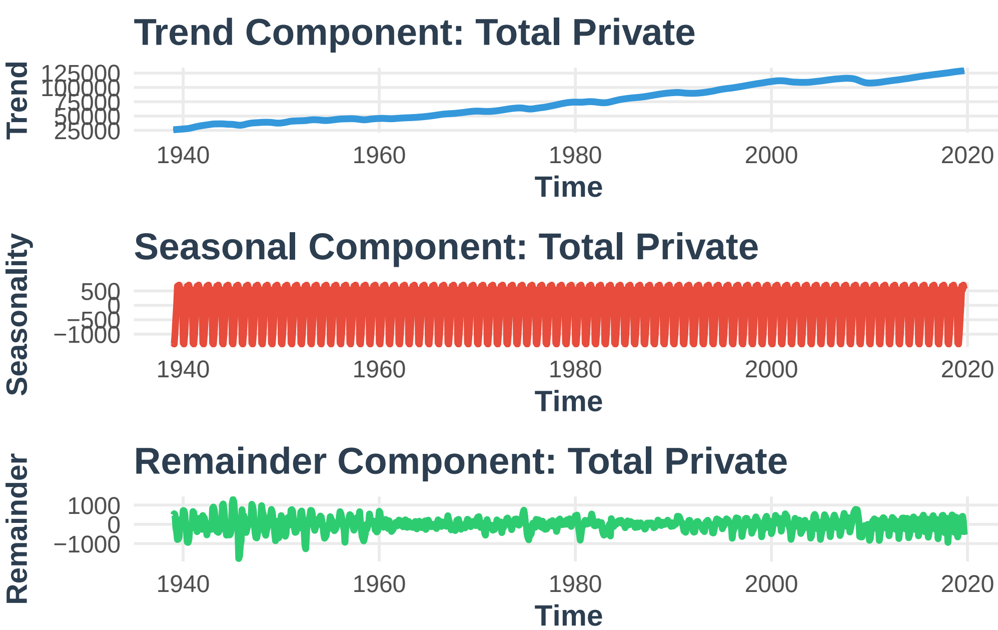
```
library(forecast)
library(ggplot2)
library(dplyr)

# Function for ARIMA forecast with colorful ribbon
plot_arima_forecast <- function(ts_data, category_name, h = 36) {
  fit <- auto.arima(ts_data)
  fc <- forecast(fit, h = h)

  # Forecast as data frame
  fc_df <- data.frame(
    Date = seq(as.Date("1939-01-01"), by = "month", length.out = length(ts_data) + h)[(length(
    Forecast = as.numeric(fc$mean),
    Lower = as.numeric(fc$lower[,2]),
    Upper = as.numeric(fc$upper[,2])
  )

  # Actual series
  actual_df <- data.frame(
    Date = seq(as.Date("1939-01-01"), by = "month", length.out = length(ts_data)),
```

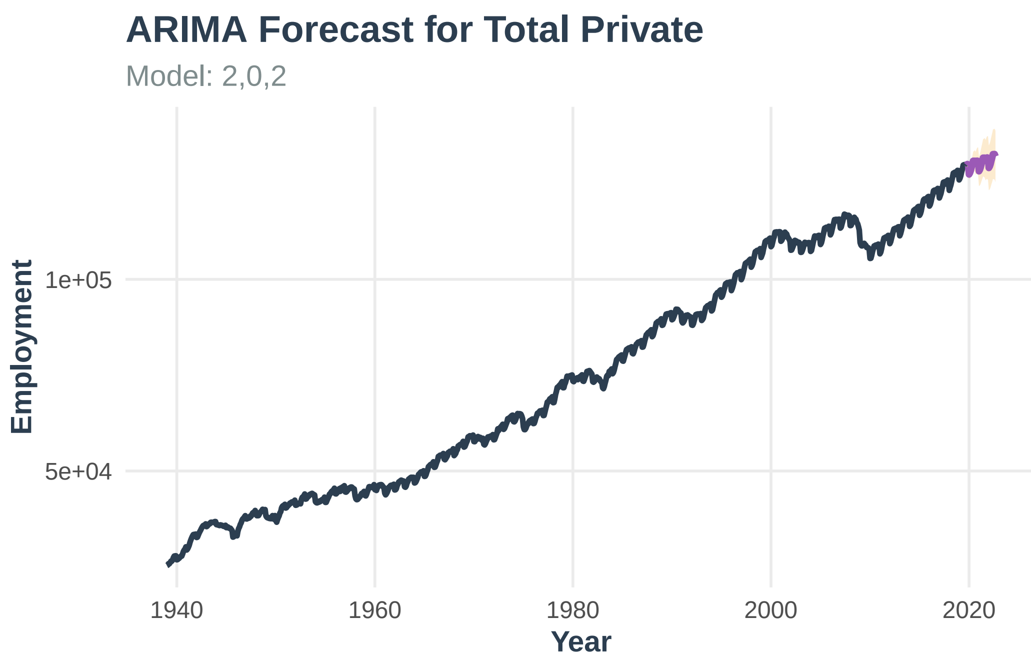
```
    Employment = as.numeric(ts_data)
  )

  ggplot() +
    geom_line(data = actual_df, aes(x = Date, y = Employment), color = "#2C3E50", size = 1) +
    geom_ribbon(data = fc_df, aes(x = Date, ymin = Lower, ymax = Upper), fill = "#F39C12", alph
    geom_line(data = fc_df, aes(x = Date, y = Forecast), color = "#9B59B6", size = 1.2) +
    labs(
      title = paste("ARIMA Forecast for", category_name),
      subtitle = paste("Model:", paste(arimaorder(fit)[1:3], collapse = ",")),
      x = "Year",
      y = "Employment"
    ) +
    professional_theme
}

# Example: Total Private
plot_arima_forecast(total_private_ts, "Total Private")
```

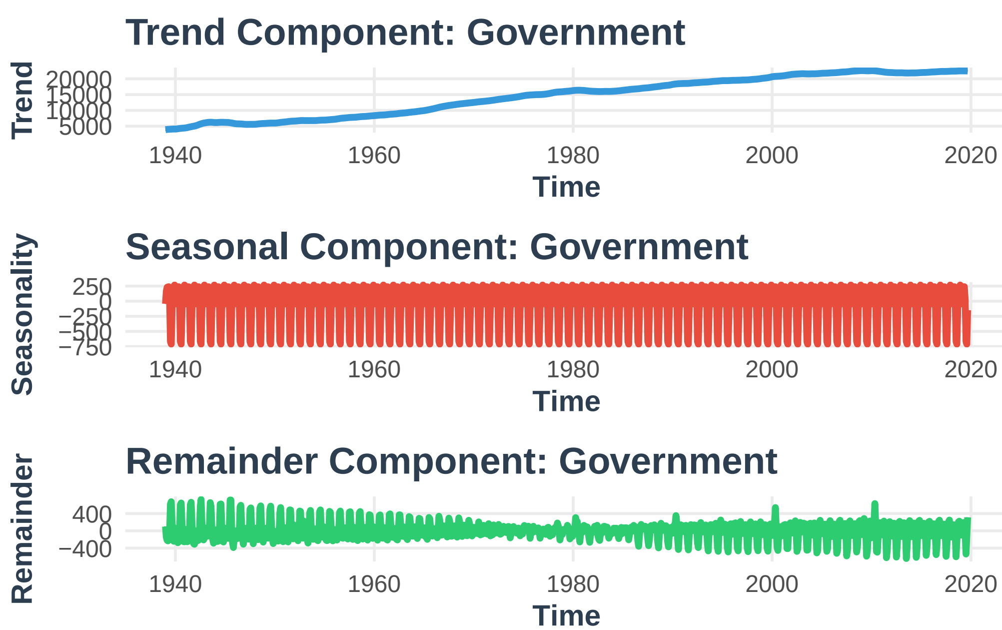


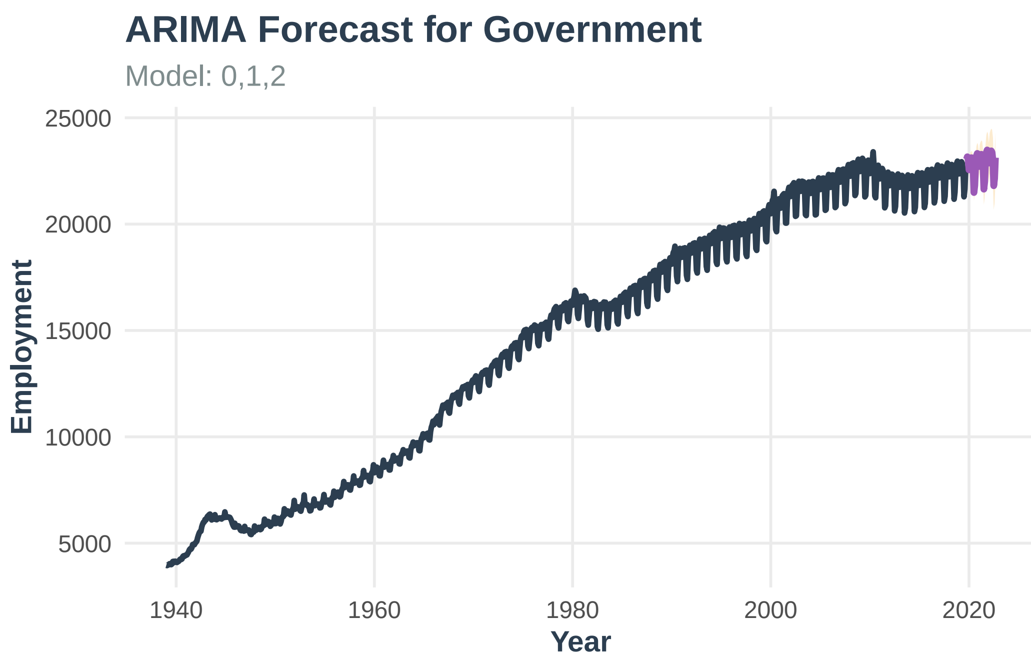
This ARIMA (2,0,2) forecast for Total Private Employment shows a strong upward trend continuing beyond 2019, indicating sustained growth in private sector employment. The purple forecast line projects gradual increases, while the light-colored shaded area represents the 95% confidence interval, showing a moderate level of uncertainty around future values. Overall, the model suggests that employment will keep rising steadily, with only minor fluctuations expected in the near future.

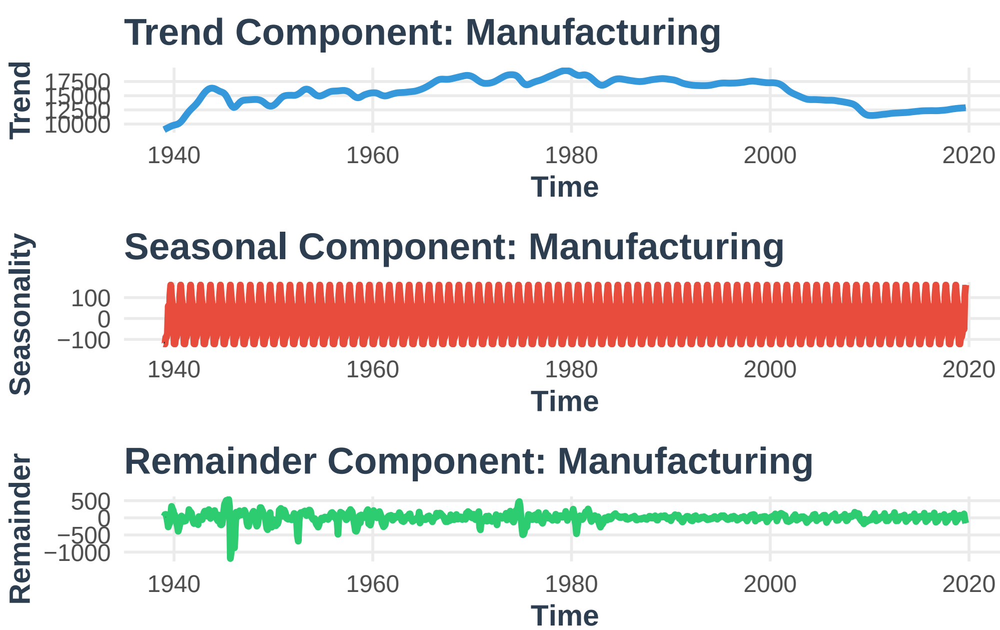
```
for(cat in key_categories) {
  ts_data <- create_employment_ts(cat)

  # STL decomposition
  stl_plot <- plot_stl_decomposition(ts_data, cat)
  print(stl_plot)

  # ARIMA forecast
  arima_plot <- plot_arima_forecast(ts_data, cat)
  print(arima_plot)
}
```

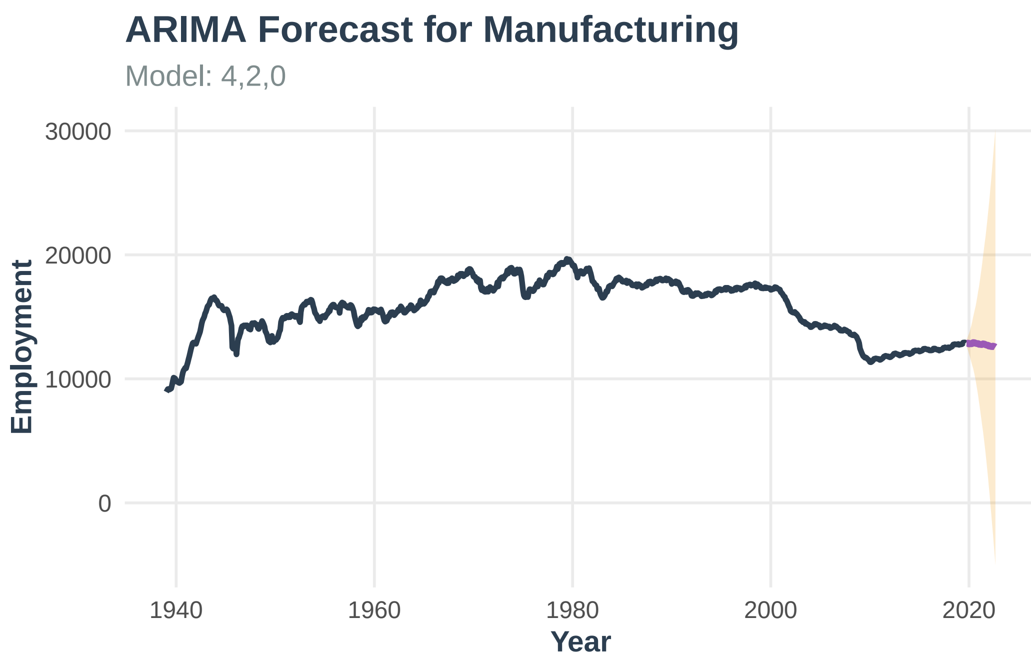


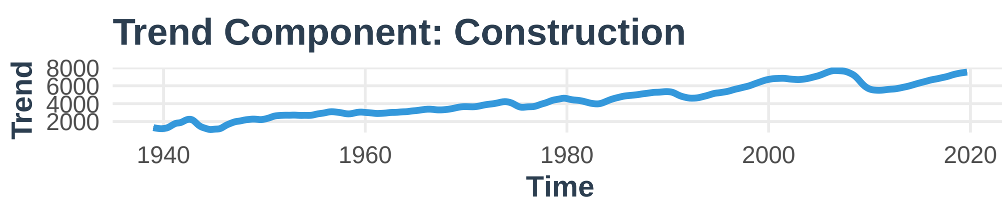

ARIMA Forecast for Total Private
Model: 2,0,2
Employment
1e+05
5e+04
1940
1960
1980
2000
2020
Year

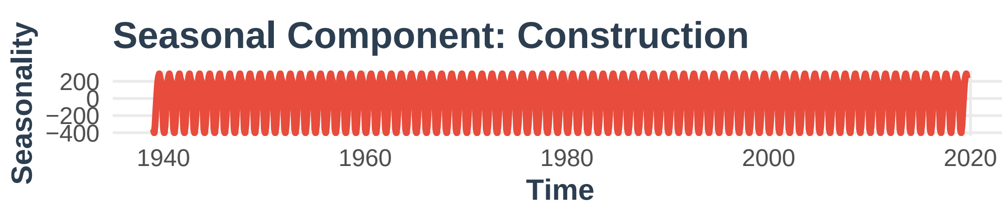

Trend Component: Government
Trend
20000
15000
10000
5000
1940
1960
1980
2000
2020
Time
Seasonal Component: Government
Seasonality
250
0
−250
−500
−750
1940
1960
1980
2000
2020
Time
Remainder Component: Government
Remainder
400
0
−400
1940
1960
1980
2000
2020
Time

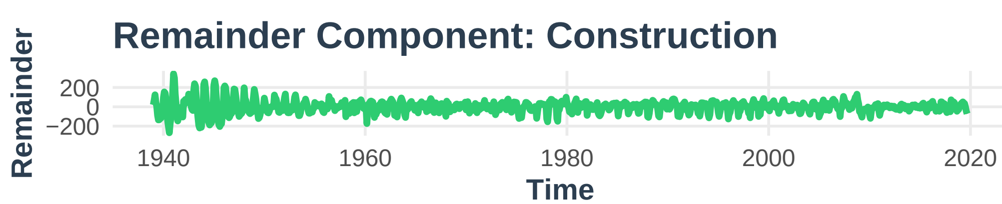

ARIMA Forecast for Government
Model: 0,1,2
Employment
25000
20000
15000
10000
5000
1940
1960
1980
2000
2020
Year

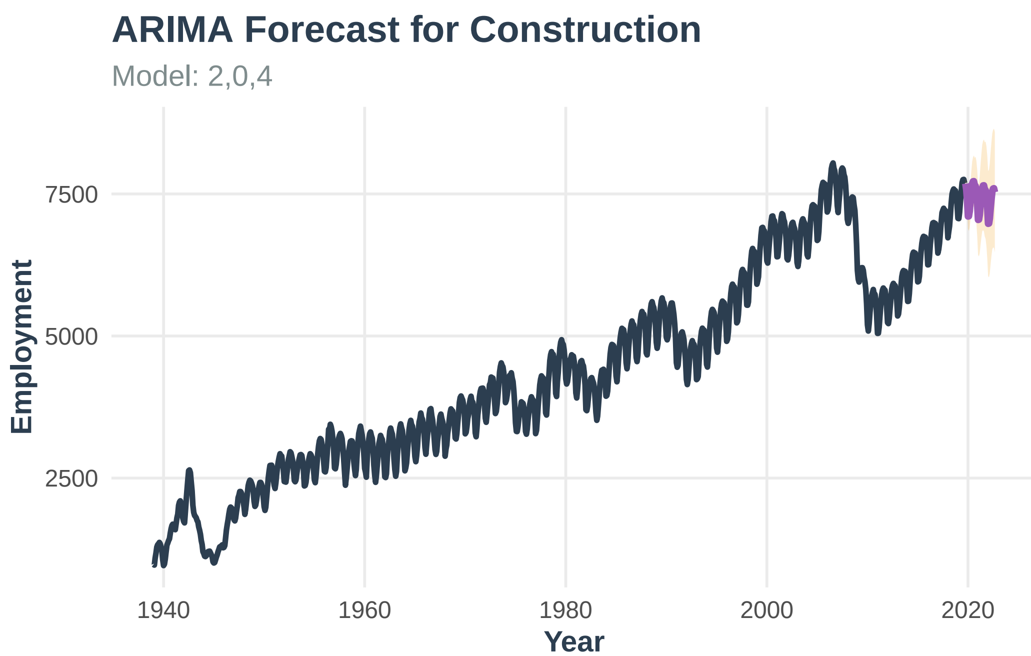

Trend Component: Manufacturing
Trend
17500
15000
12500
10000
1940
1960
1980
2000
2020
Time
Seasonal Component: Manufacturing
Seasonality
100
0
−100
1940
1960
1980
2000
2020
Time
Remainder Component: Manufacturing
Remainder
500
0
−500
−1000
1940
1960
1980
2000
2020
Time

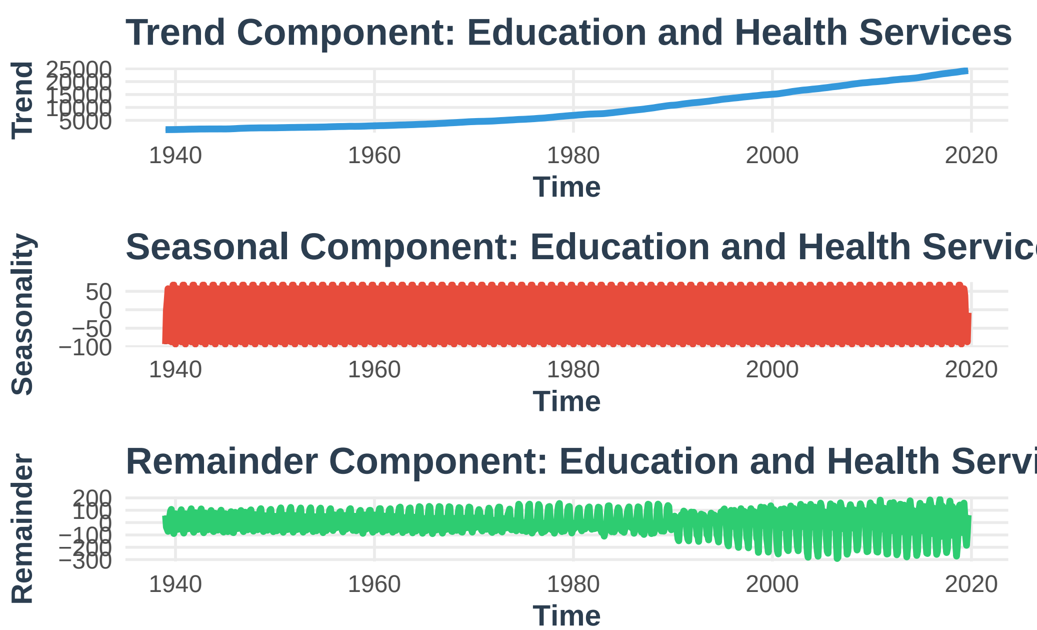

ARIMA Forecast for Manufacturing
Model: 4,2,0
Employment
30000
20000
10000
0
1940
1960
1980
2000
2020
Year

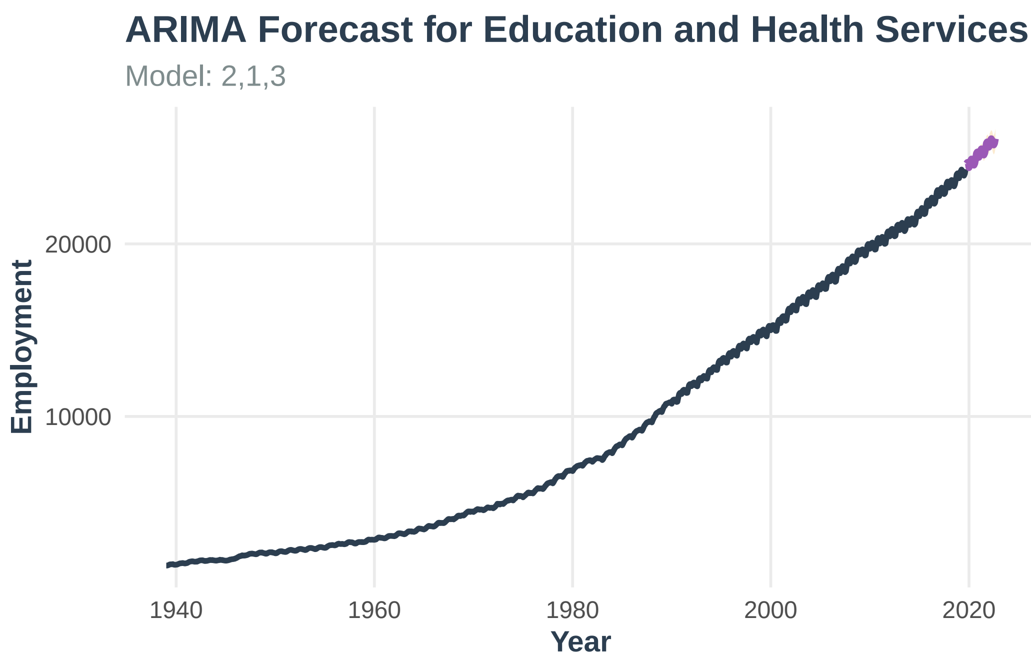

Trend Component: Construction
Trend
8000
6000
4000
2000
1940
1960
1980
2000
2020
Time

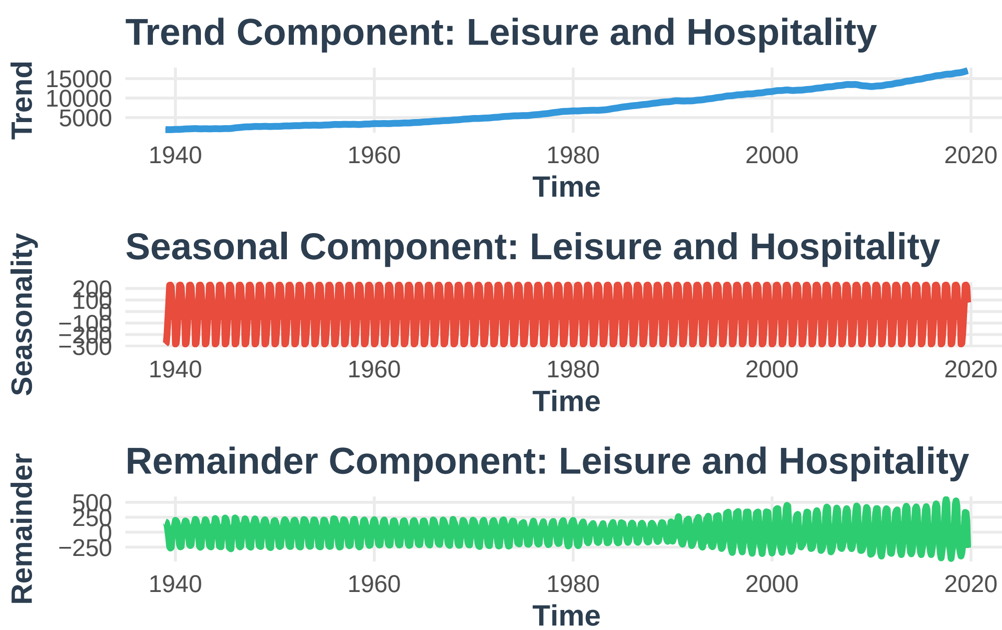

Seasonal Component: Construction
Seasonality
200
0
−200
−400
1940
1960
1980
2000
2020
Time

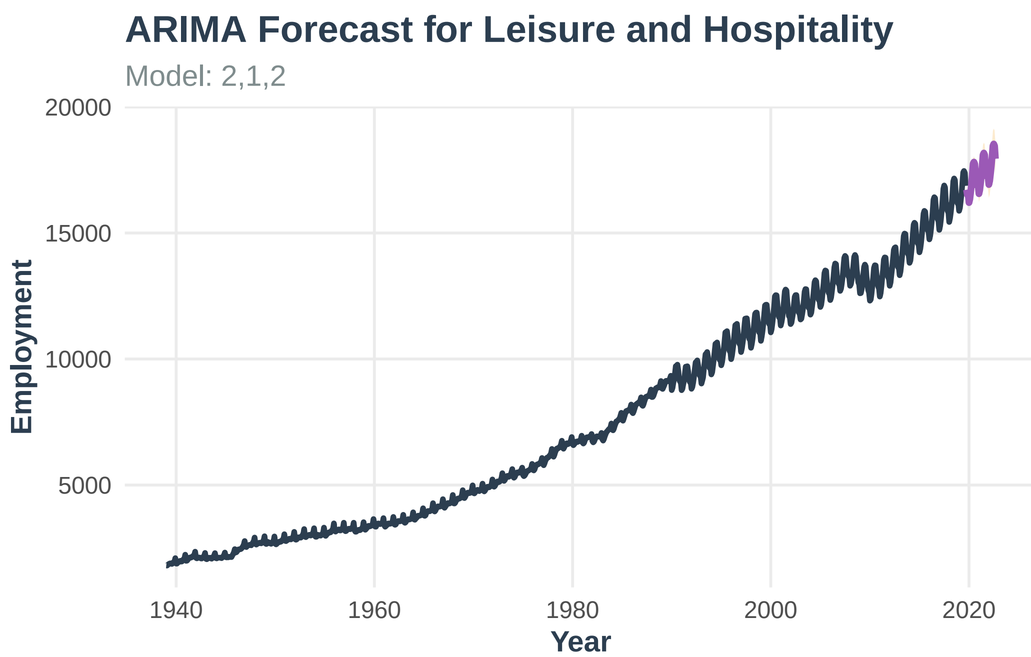

Remainder Component: Construction
Remainder
200
0
−200
1940
1960
1980
2000
2020
Time


ARIMA Forecast for Construction
Model: 2,0,4
Employment
7500
5000
2500
1940
1960
1980
2000
2020
Year


Trend Component: Education and Health Services
Trend
Time
Seasonal Component: Education and Health Servic
Seasonality
50
0
−50
−100
Time
Remainder Component: Education and Health Servi
Remainder
Time
1940
1960
1980
2000
2020


ARIMA Forecast for Education and Health Services
Model: 2,1,3
Employment
20000
10000
1940
1960
1980
2000
2020
Year


Trend Component: Leisure and Hospitality
Trend
15000
10000
5000
1940
1960
1980
2000
2020
Time
Seasonal Component: Leisure and Hospitality
Seasonality
200
−300
1940
1960
1980
2000
2020
Time
Remainder Component: Leisure and Hospitality
Remainder
500
250
0
−250
1940
1960
1980
2000
2020
Time



The overall analysis of U.S. employment sectors shows strong long-term growth across most industries, with distinct trends and seasonal behaviors. The Total Private and Construction sectors display steady upward trends with noticeable seasonal patterns and short-term fluctuations, indicating cyclical but consistent job growth. Government employment increased gradually and stabilized in recent decades, showing moderate seasonality and steady public-sector demand. Manufacturing peaked around the mid-20th century and has since declined, with forecasts suggesting stable but low employment levels, reflecting industrial shifts. In contrast, Education and Health Services show continuous and strong growth, highlighting the sector's resilience and expanding workforce needs. Finally, Leisure and Hospitality reveal steady long-term growth with strong seasonality linked to tourism cycles, and forecasts suggest further moderate increases. Overall, the time series and ARIMA forecasts indicate that while traditional sectors like manufacturing have plateaued, service-oriented and construction industries continue to drive employment growth in the U.S.

`The END`