FORECASTING USING R

# Dynamic regression

Rob Hyndman
Author, forecast

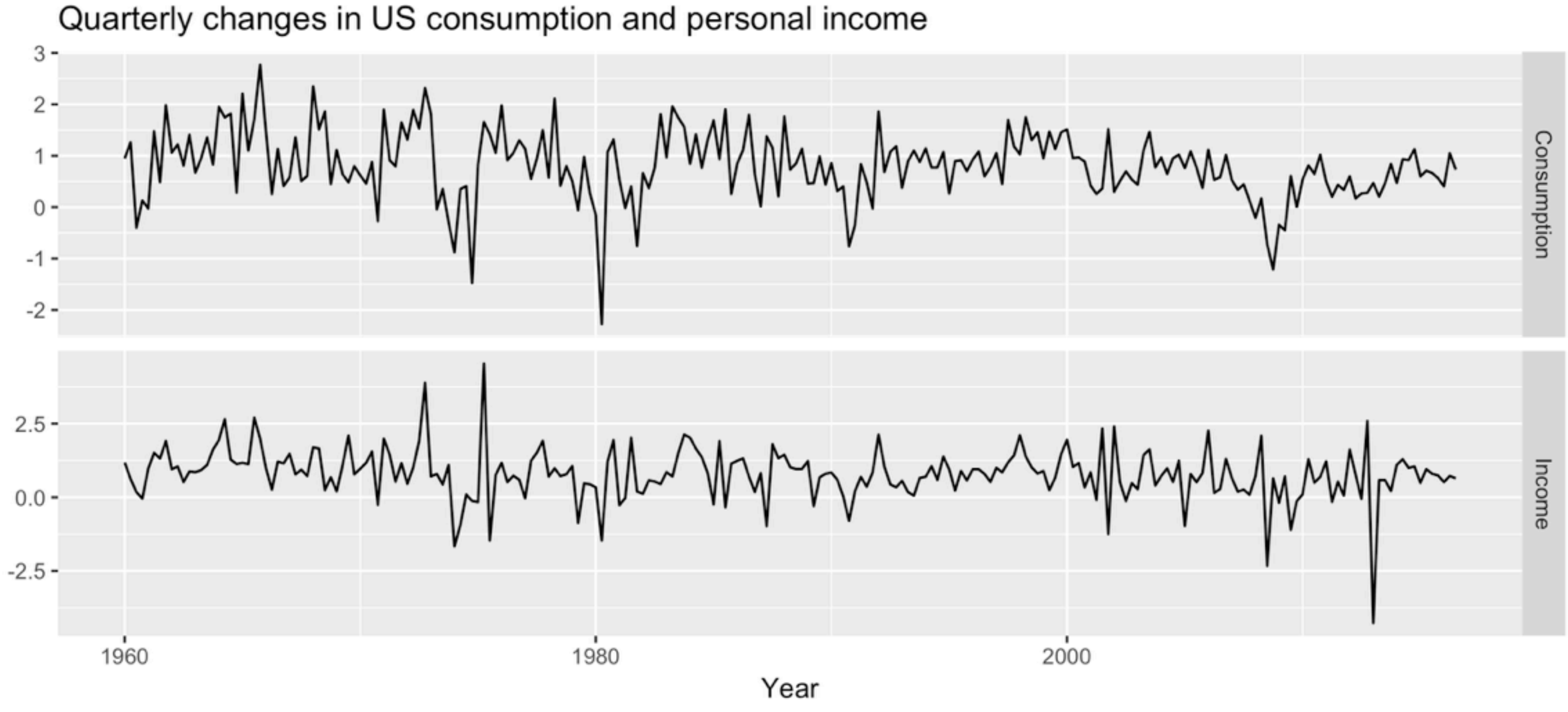
# Dynamic regression

| Regression model with ARIMA errors |
| --- |
| $y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_r x_{r,t} + e_t$ |

- $y_t$ modeled as function of $r$ explanatory variables $x_{1,t}, \ldots, x_{r,t}$
- In dynamic regression, we allow $e_t$ to be an ARIMA process
- In ordinary regression, we assume that $e_t$ is white noise

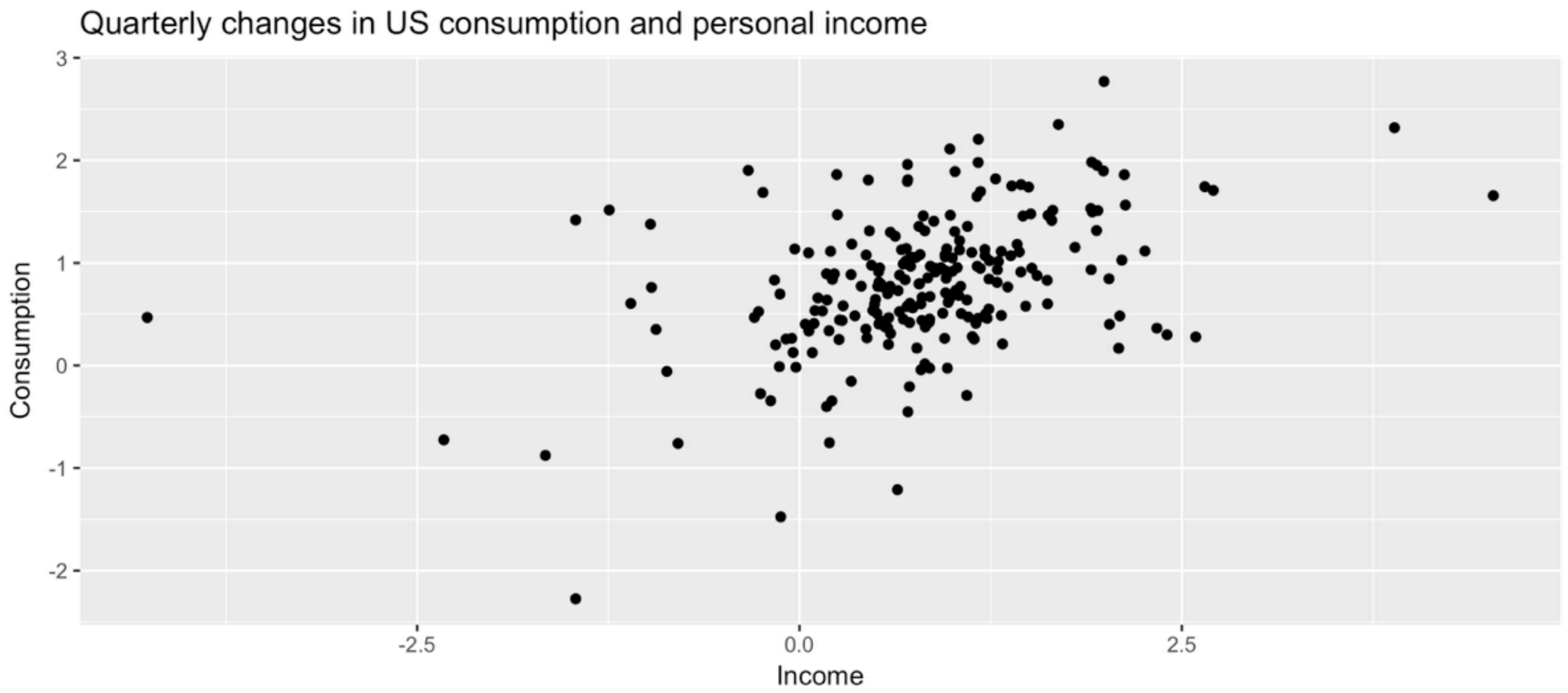
# US personal consumption and income

```
> autoplot(uschange[,1:2], facets = TRUE) +
    xlab("Year") + ylab("") +
    ggtitle("Quarterly changes in US consumption
            and personal income")
```



# US personal consumption and income

```
> ggplot(aes(x = Income, y = Consumption),
         data = as.data.frame(uschange)) +
    geom_point() +
    ggtitle("Quarterly changes in US consumption and
            personal income")
```

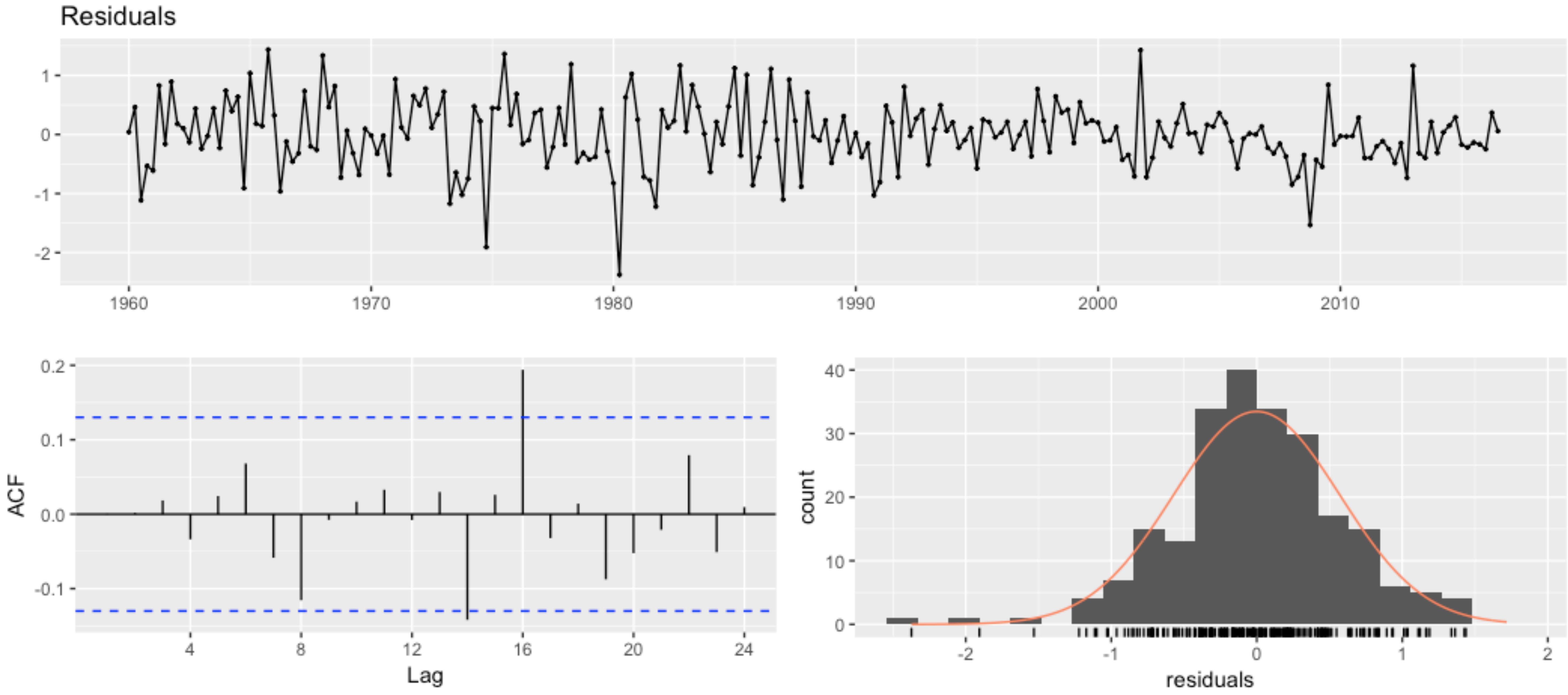


# Dynamic regression model for US personal consumption

```
> fit <- auto.arima(uschange[,"Consumption"],
                    xreg = uschange[,"Income"])
> fit

Series: uschange[, "Consumption"]
Regression with ARIMA(1,0,2) errors

Coefficients:
         ar1      ma1     ma2  intercept  origxreg
      0.6191  -0.5424  0.2367     0.6099    0.2492
s.e.  0.1422   0.1475  0.0685     0.0777    0.0459

sigma^2 estimated as 0.334:  log likelihood=-195.22
AIC=402.44   AICc=402.82   BIC=422.99
```

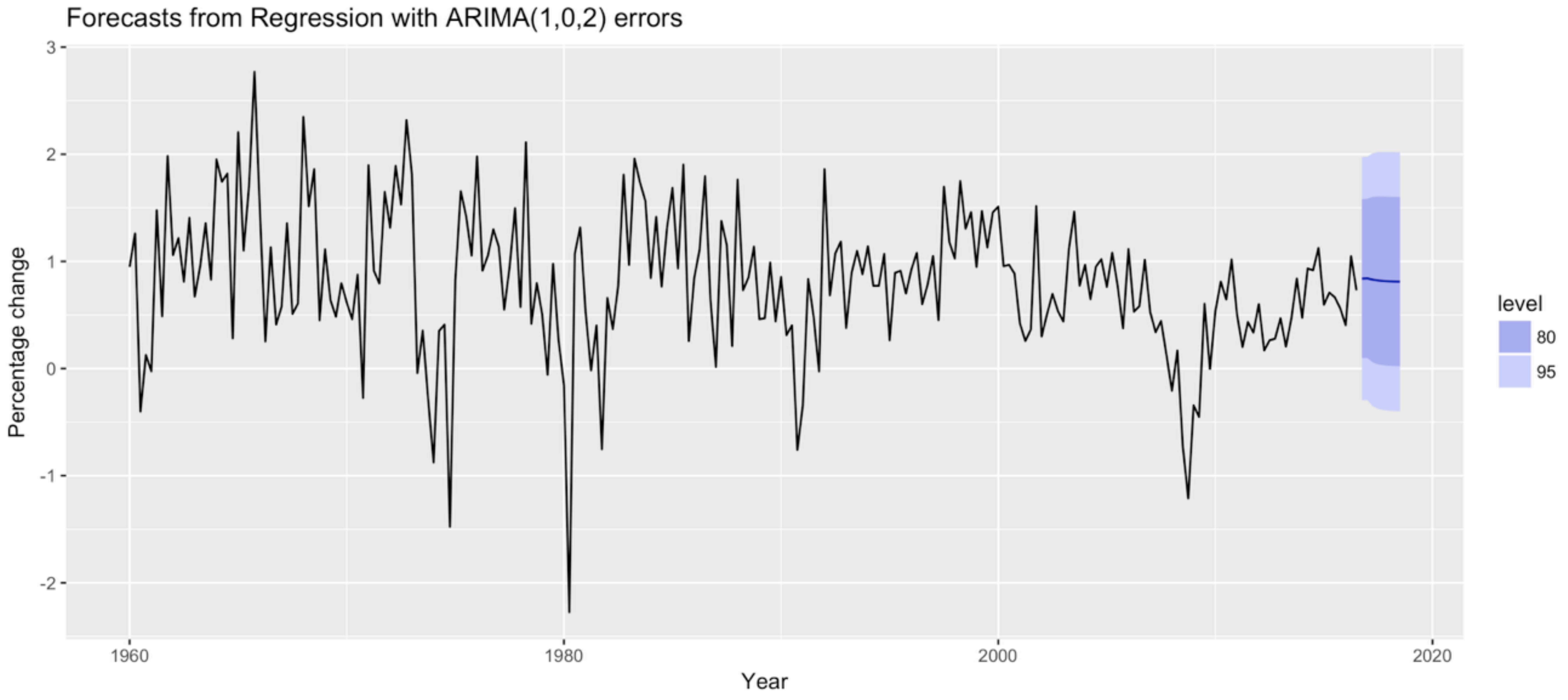
# Residuals from dynamic regression model

```
> checkresiduals(fit)
  Ljung-Box test
data:  residuals
Q* = 5.5543, df = 3, p-value = 0.1354
Model df: 5.   Total lags used: 8
```



# Forecasts from dynamic regression model

```
> fcast <- forecast(fit, xreg = rep(0.8, 8))
> autoplot(fcast) +
    xlab("Year") + ylab("Percentage change")
```



# Let's practice!

# Dynamic harmonic regression

# Dynamic harmonic regression

Periodic seasonality can be handled using pairs of Fourier terms:

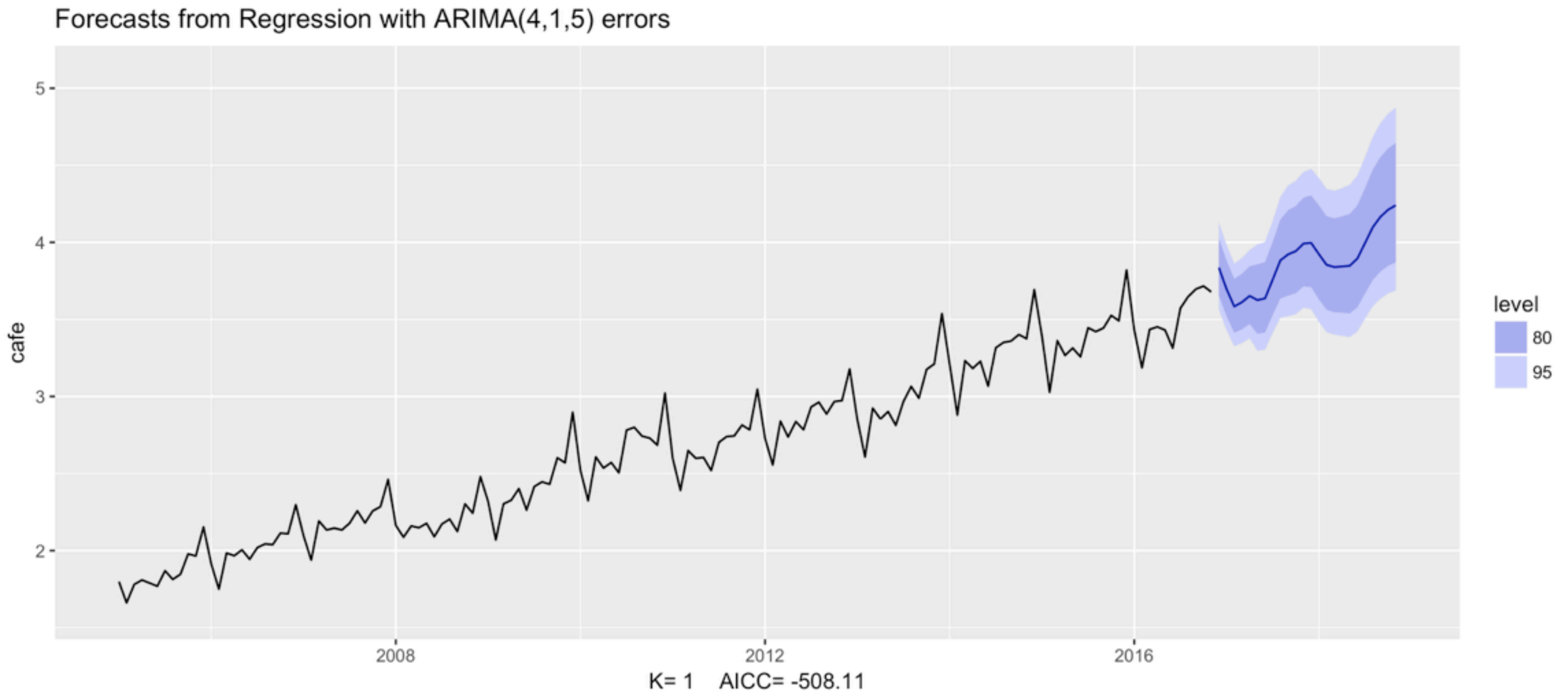
$$s_k(t) = \sin\left(\frac{2\pi kt}{m}\right) \qquad c_k(t) = \cos\left(\frac{2\pi kt}{m}\right)$$

$$\boxed{y_t = \beta_0 + \sum_{k=1}^{K}\left[\alpha_k s_k(t) + \gamma_k c_k(t)\right] + e_t}$$

- $m =$ seasonal period
- Every periodic function can be approximated by sums of sin and cos terms for large enough $K$
- Regression coefficients: $\alpha_k$ and $\gamma_k$
- $e_t$ can be modeled as a non-seasonal ARIMA process
- Assumes seasonal pattern is unchanging

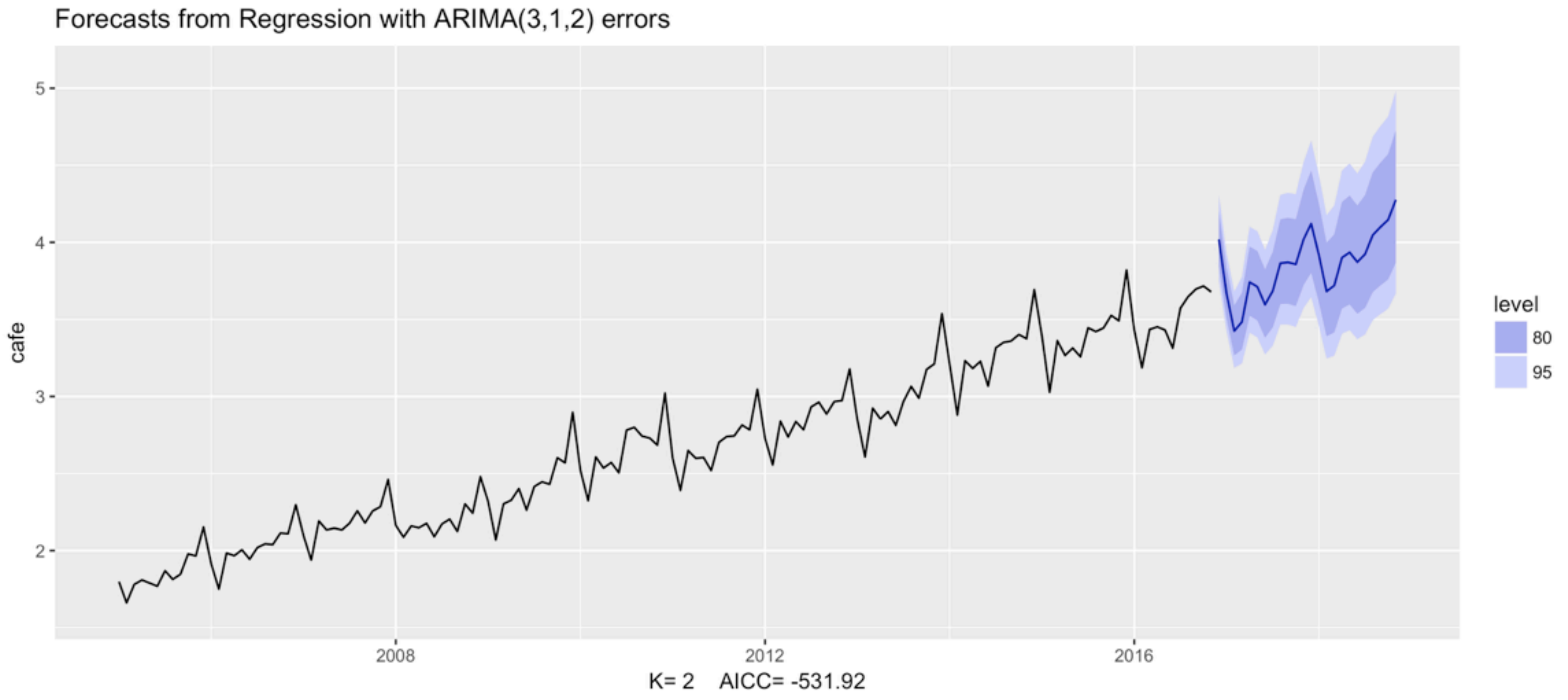
# Example: Australian cafe expenditure

```
> fit <- auto.arima(cafe, xreg = fourier(cafe, K = 1),
                    seasonal = FALSE, lambda = 0)
> fit %>% forecast(xreg = fourier(cafe, K = 1, h = 24)) %>%
    autoplot() + ylim(1.6, 5.1)
```



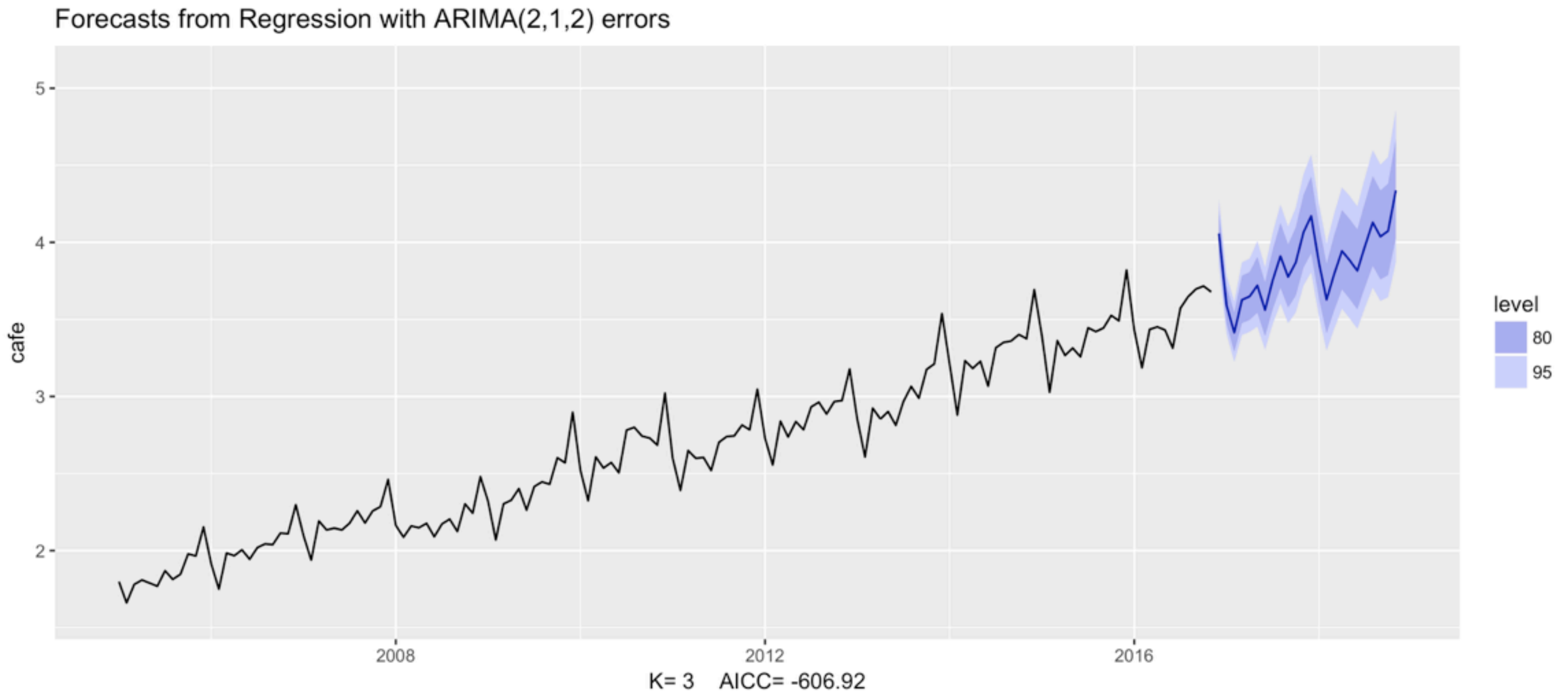
# Example: Australian cafe expenditure

```
> fit <- auto.arima(cafe, xreg = fourier(cafe, K = 2),
                     seasonal = FALSE, lambda = 0)
> fit %>% forecast(xreg = fourier(cafe, K = 2, h = 24)) %>%
    autoplot() + ylim(1.6, 5.1)
```



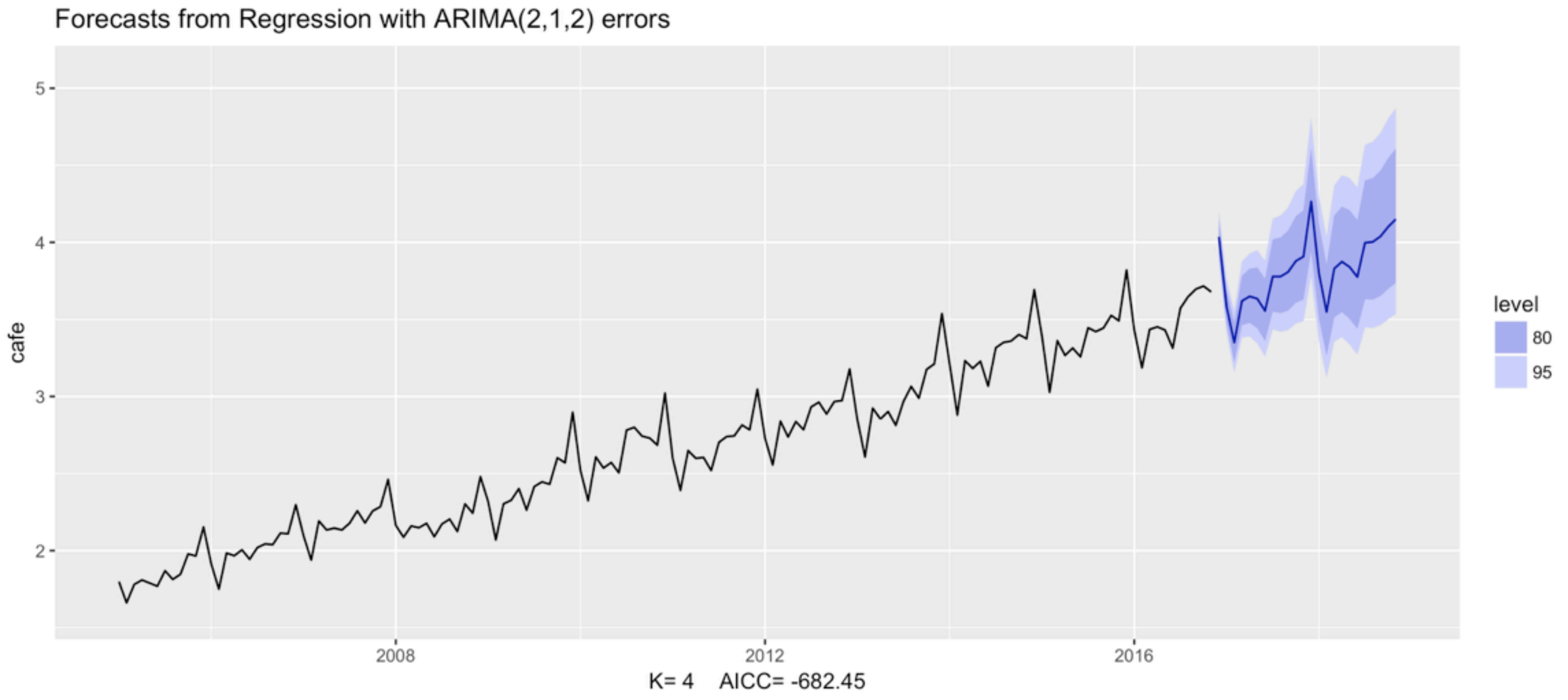
# Example: Australian cafe expenditure

```
> fit <- auto.arima(cafe, xreg = fourier(cafe, K = 3),
                    seasonal = FALSE, lambda = 0)
> fit %>% forecast(xreg = fourier(cafe, K = 3, h = 24)) %>%
    autoplot() + ylim(1.6, 5.1)
```



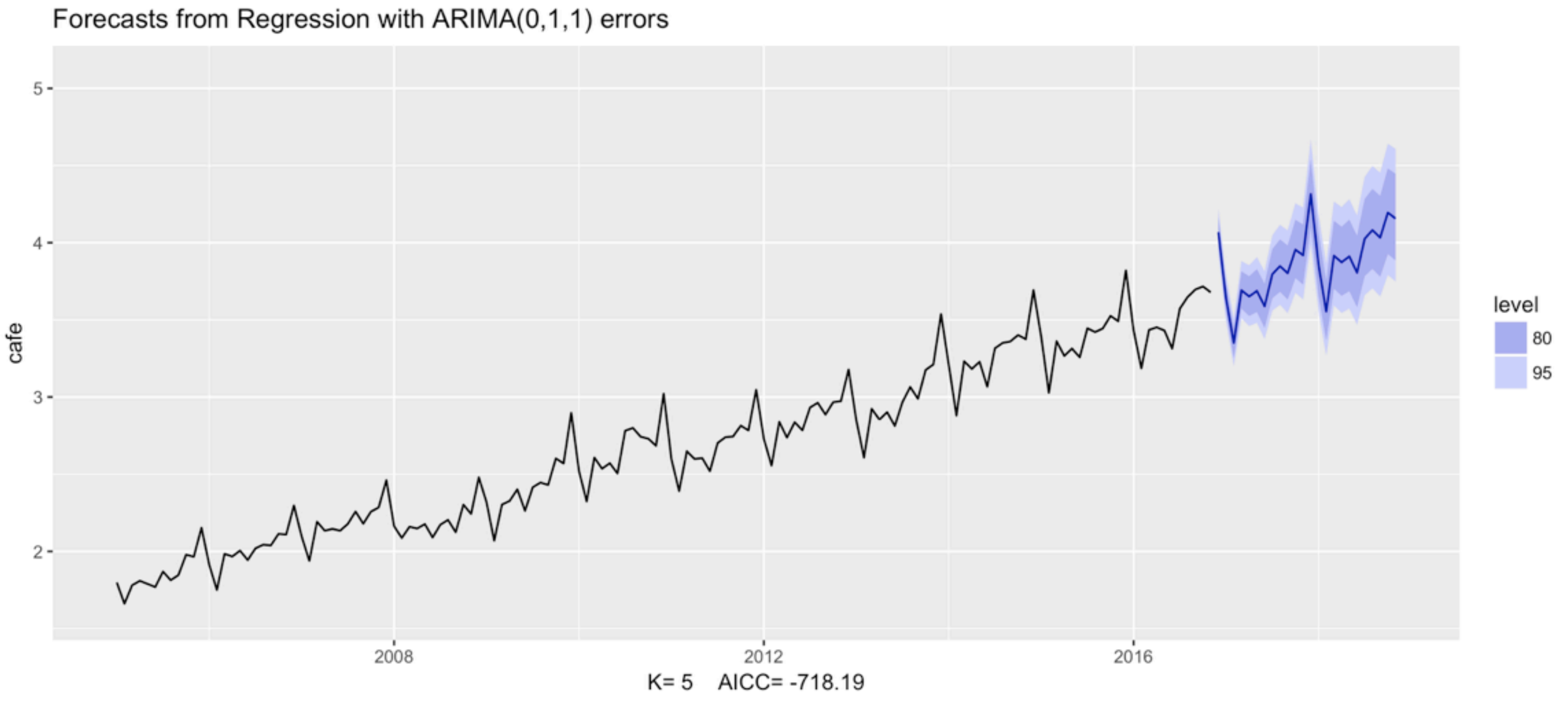
# Example: Australian cafe expenditure

```
> fit <- auto.arima(cafe, xreg = fourier(cafe, K = 4),
                         seasonal = FALSE, lambda = 0)
> fit %>% forecast(xreg = fourier(cafe, K = 4, h = 24)) %>%
      autoplot() + ylim(1.6, 5.1)
```



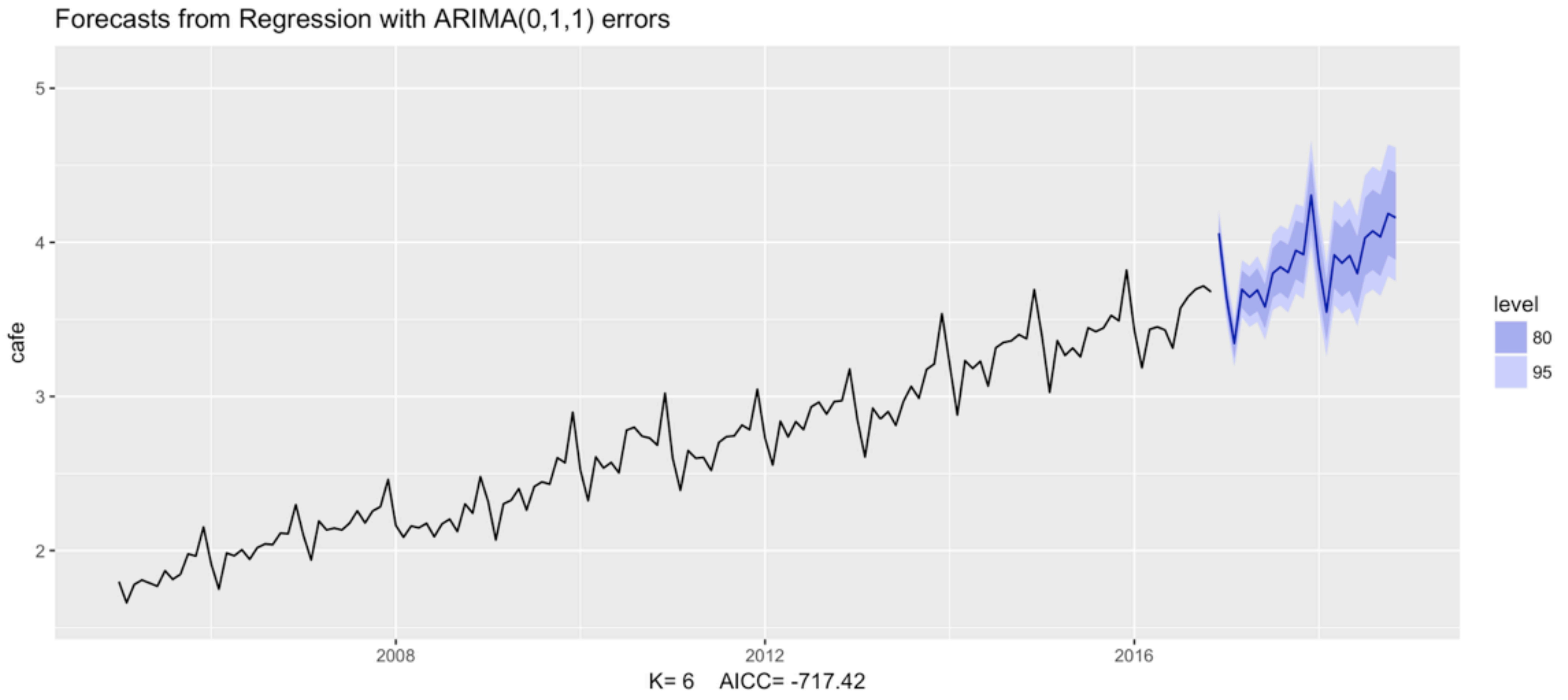
# Example: Australian cafe expenditure

```
> fit <- auto.arima(cafe, xreg = fourier(cafe, K = 5),
                    seasonal = FALSE, lambda = 0)
> fit %>% forecast(xreg = fourier(cafe, K = 5, h = 24)) %>%
    autoplot() + ylim(1.6, 5.1)
```



# Example: Australian cafe expenditure

```
> fit <- auto.arima(cafe, xreg = fourier(cafe, K = 6),
                    seasonal = FALSE, lambda = 0)
> fit %>% forecast(xreg = fourier(cafe, K = 6, h = 24)) %>%
    autoplot() + ylim(1.6, 5.1)
```



# Dynamic harmonic regression

$$y_t = \beta_0 + \beta_1 x_{t,1} + \cdots + \beta_{t,r} x_{t,r} + \sum_{k=1}^{K} [\alpha_k s_k(t) + \gamma_k c_k(t)] + e_t$$

- Other predictor variables can be added as well: $x_{t,1}, \ldots, x_{t,r}$
- Choose $K$ to minimize the AICc
- $K$ can not be more than $m/2$
- This is particularly useful for weekly data, daily data and sub-daily data

# Let's practice!

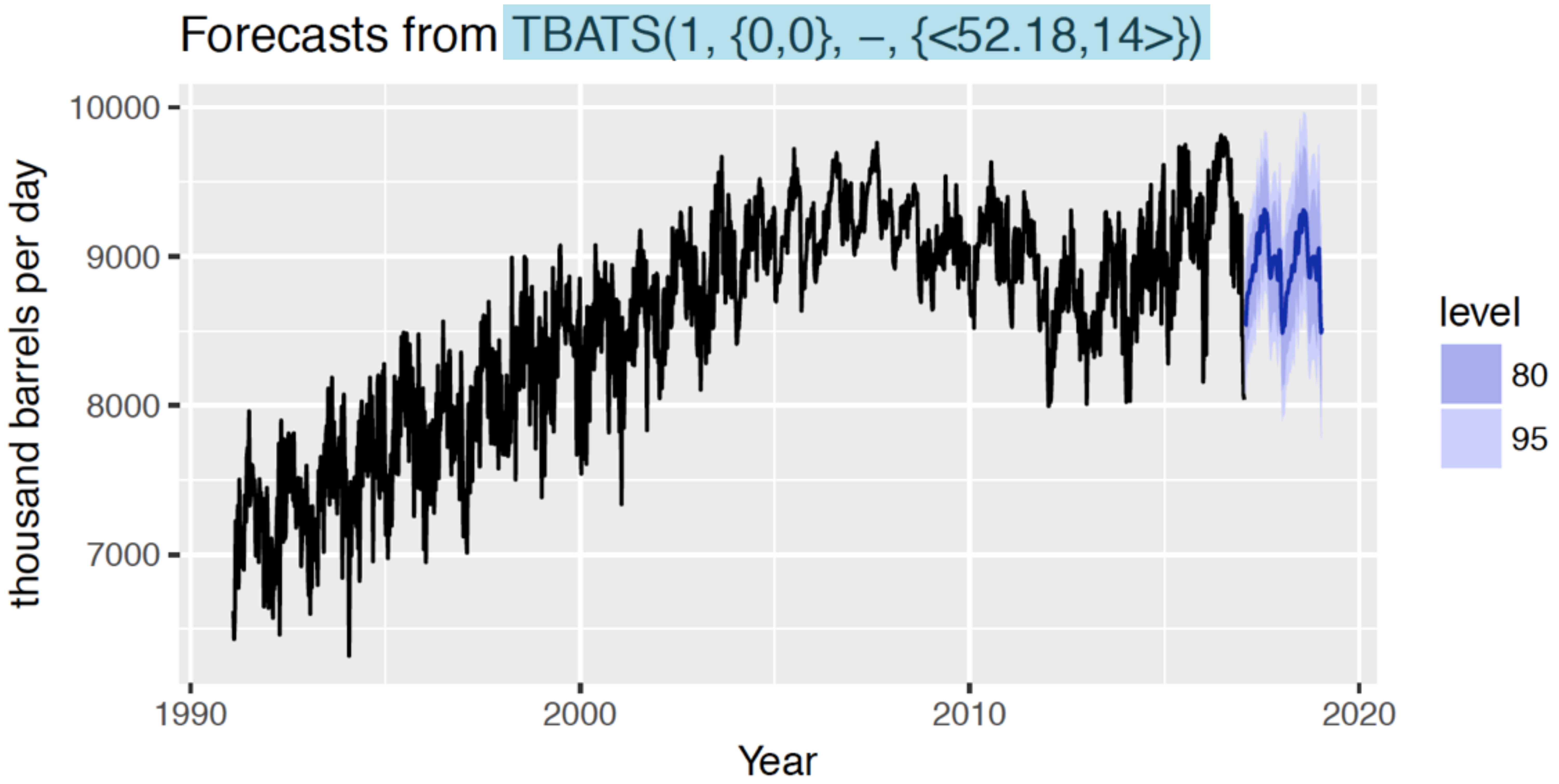
# TBATS models

# TBATS model

- Trigonometric terms for seasonality
- Box-Cox transformations for heterogeneity
- ARMA errors for short-term dynamics
- Trend (possibly damped)
- Seasonal (including multiple and non-integer periods)

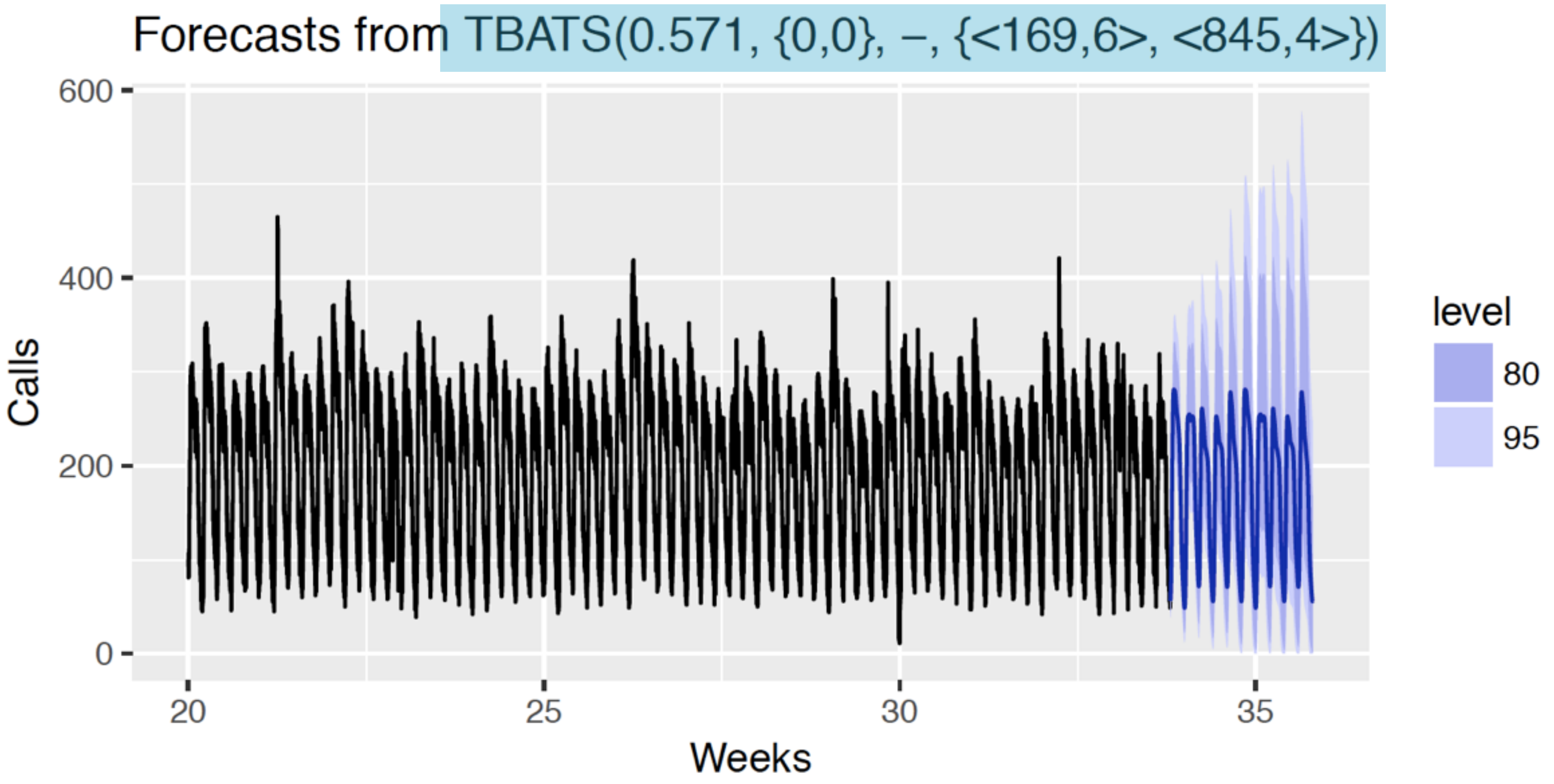
# US Gasoline data

```
> gasoline %>% tbats() %>% forecast() %>%
    autoplot() +
    xlab("Year") + ylab("thousand barrels per day")
```



# Call center data

```
> calls %>% window(start = 20) %>%
    tbats() %>% forecast() %>%
    autoplot() + xlab("Weeks") + ylab("Calls")
```



# TBATS model

- Trigonometric terms for seasonality
- Box-Cox transformations for heterogeneity
- ARMA errors for short-term dynamics
- Trend (possibly damped)
- Seasonal (including multiple and non-integer periods)
    - Handles non-integer seasonality, multiple seasonal periods
    - Entirely automated
    - Prediction intervals often too wide
    - Very slow on long series

# Let's practice!