Impulse Responses by Local Projections

Yan Bo Zeng

February 29, 2020

```
library(readr)
library(readxl)
library(dplyr)
library(tidyr)
library(sandwich)
```

Data Handling

```
## Read in datasets
ford_tfp <- read_csv("ford_tfp.csv")
quarterly_tfp <- read_excel("quarterly_tfp.xlsx", sheet = "quarterly",
                            skip = 1)[, c("date","dtfp_util")]
ie_data <- read_excel("ie_data.xls", sheet = "Data", skip = 7)[, 1:2]

CNP16OV <- read_csv("CNP16OV.csv")
GDP <- read_csv("GDP.csv")
GDPDEF <- read_csv("GDPDEF.csv")
HOANBS <- read_csv("HOANBS.csv")
PCE <- read_csv("PCE.csv")
PNFI <- read_csv("PNFI.csv")

## Time Series Conversion + Name change
## a frequency of 4 indicates a quarterly series
ford_tfp <- ts(ford_tfp$ford_tfp, start = 1949, frequency = 4)
d_tfp <- ts(quarterly_tfp$dtfp_util, start = 1947, frequency = 4)
## a frequency of 12 indicates a monthly series
stock_prices <- ts(ie_data$P, start = 1871, frequency = 12)

population <- ts(CNP16OV$CNP16OV, start = 1948, frequency = 12)
GDP <- ts(GDP$GDP, start = 1947, frequency = 4)
GDP_deflator <- ts(GDPDEF$GDPDEF, start = 1947, frequency = 4)
hours <- ts(HOANBS$HOANBS, start = 1947, frequency = 4)
consumption <- ts(PCE$PCE, start = 1959, frequency = 12)
investment <- ts(PNFI$PNFI, start = 1947, frequency = 4)

## Convert monthly datasets into quarterly average sets
stock_prices <- aggregate(stock_prices, nfrequency = 4,
                          FUN = mean) %>% window(start = 1947)
population <- aggregate(population, nfrequency = 4, FUN = mean)
consumption <- aggregate(consumption, nfrequency = 4, FUN = mean)

## Merge datasets
```

```r
data <- ts.union(ford_tfp, d_tfp, stock_prices, population,
                 GDP, GDP_deflator, hours, consumption, investment)
```

Data Processing

```r
## compute a log real per-capita version of GDP, consumption,
## non-residential investment, and stock prices
## Multiply each series by 100 so that it is expressed in percent.
denominator <- data[, "population"] * data[, "GDP_deflator"]

ln_re_pc_GDP <- log(data[, "GDP"] / denominator) * 100
ln_re_pc_consumption <- log(data[, "consumption"] / denominator) * 100
ln_re_pc_investment <- log(data[, "investment"] / denominator) * 100
ln_re_pc_stock_prices <- log(data[, "stock_prices"] / denominator) * 100

## Compute log hours per capita by dividing hours by population,
## taking logs, and multiplying by 100.
ln_pc_hours <- log(data[, "hours"] / data[, "population"]) * 100

## Compute log real labor productivity by dividing GDP by hours and the
## GDP deflator, taking logs, and multiplying by 100
denominator <- data[, "hours"] * data[, "GDP_deflator"]
ln_re_lab_prod <- log(data[, "GDP"] / denominator) * 100
```

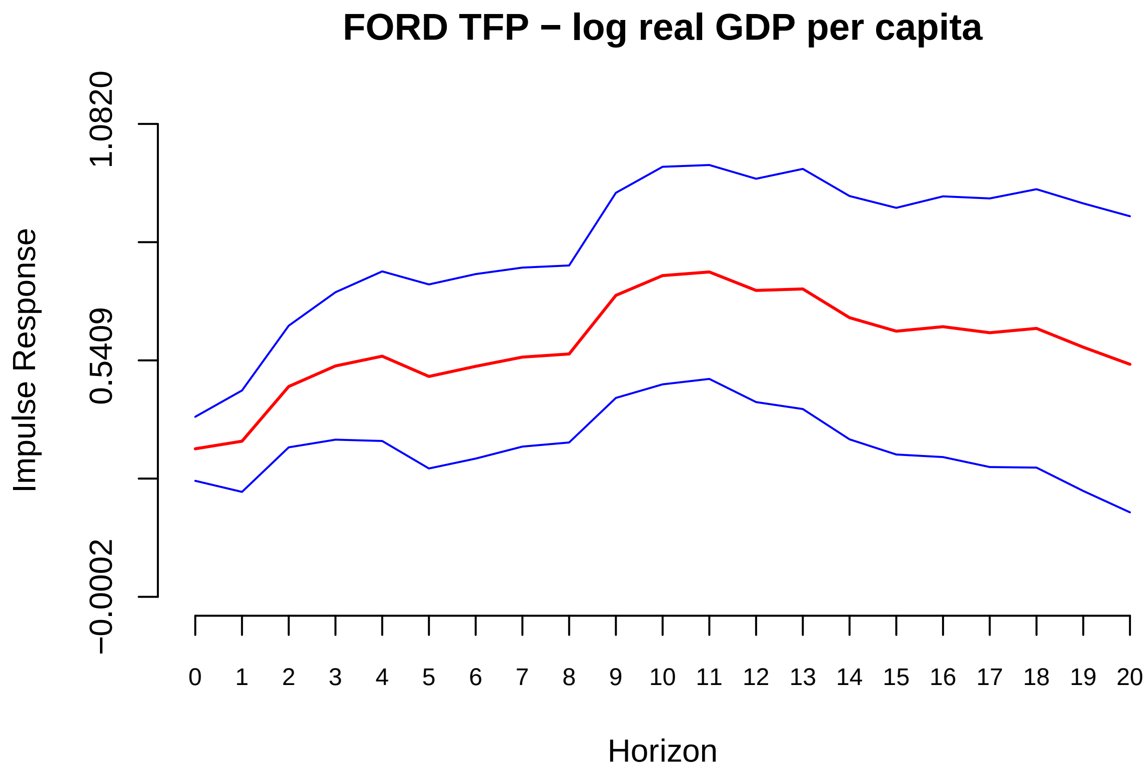
Local Projection Function

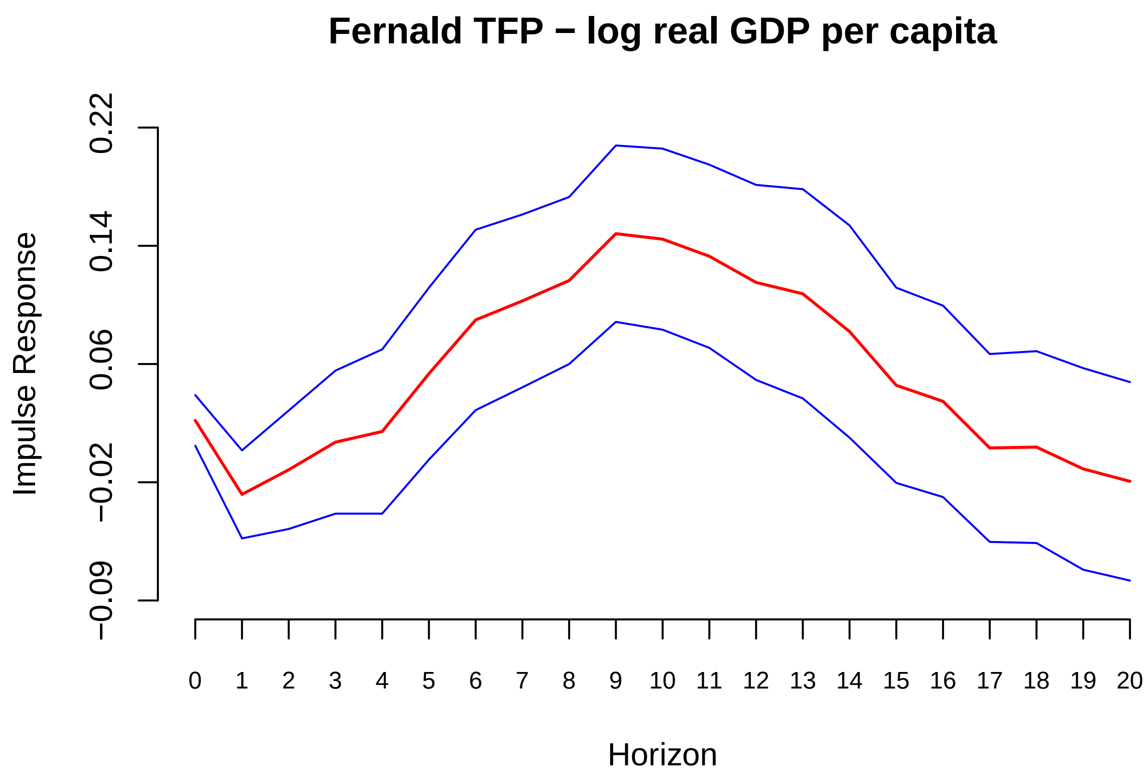
```r
## Please only use time series with correctly matched time stamps.

lp <- function(depend_var, shock_var, control1, control2, control3, h) {

        outcome <- stats::lag(depend_var, h) ## outcome ahead by h horizons

        shock1 <- stats::lag(shock_var, -1) ## shock - lag 1
        shock2 <- stats::lag(shock_var, -2) ## shock - lag 2
        control1_lag <- stats::lag(control1, -1) ## control variable 1 - lag 1
        control2_lag <- stats::lag(control2, -1) ## control variable 2 - lag 1
        depend_lag <- stats::lag(depend_var, -1) ## dependent - lag 1

        if(missing(control3)) {
                model_mat <- ts.union(outcome, shock_var, shock1, shock2,
                                      depend_lag, control1_lag, control2_lag)
                return(lm(outcome~., data = model_mat))
        }

        control3_lag <- stats::lag(control3, -1) ## control variable 3 - lag 1

        model_mat <- ts.union(outcome, shock_var, shock1, shock2, depend_lag,
                              control1_lag, control2_lag, control3_lag)

        return(lm(outcome~., data = model_mat))
}
```

Plot Function

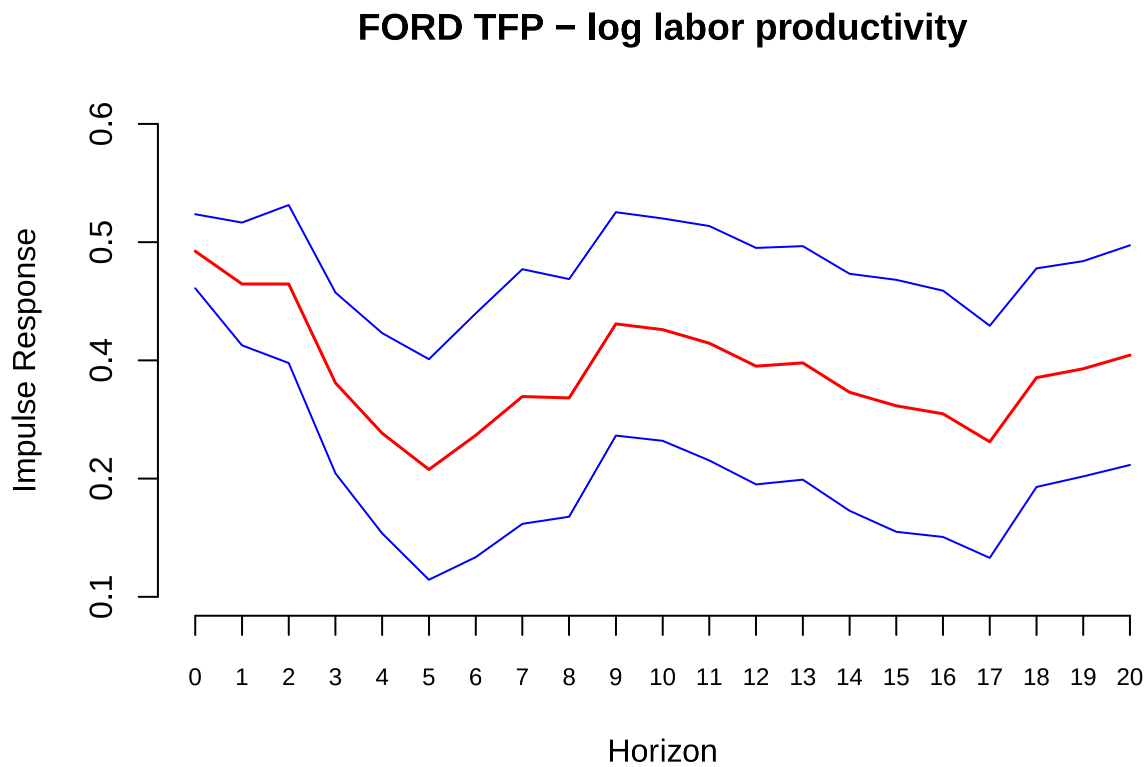
```
plot_lp <- function(depend_var, shock_var, control1, control2,
                    control3, h = 20, conf_level = 0.667, main) {
        results <- 0:h; HAC <- 0:h

        ## If there is NOT a 3rd control variable
        if(missing(control3)) {
                for(i in 0:h) {
                        model <- lp(depend_var, shock_var, control1,
                                    control2, h = i)
                        ## Coefficient for current shock
                        results[i+1] <- model$coefficients[2]
                        ## HAC standard errors at each horizon
                        HAC[i+1] <- NeweyWest(model, lag = i,
                                              prewhite = FALSE)[2,2] %>% sqrt()
                }
                low <- min(results)+qnorm((1-conf_level)/2)*max(HAC)
                up <- max(results)-qnorm((1-conf_level)/2)*max(HAC)

                plot(results, main = main, type = "l", col = "red",
                     lwd = 1.5, ylab = "Impulse Response", xlab = "Horizon",
                     ylim = c(low, up), axes = FALSE)

                axis(1, at = 1:(h+1), labels = 0:h, cex.axis = 0.75)
                axis(2, at = seq(low, up, (up-low)/4),
                     labels = format(seq(low, up, (up-low)/4),
                                     digits = 1, scientific = FALSE))
                lines(results + qnorm((1-conf_level)/2) * HAC, col = "blue")
                lines(results - qnorm((1-conf_level)/2) * HAC, col = "blue")

        } else { ## If there is a 3rd control variable
                for(i in 0:h) {
                        model <- lp(depend_var, shock_var, control1, control2,
                                    control3, h = i)
                        results[i+1] <- model$coefficients[2]
                        HAC[i+1] <- NeweyWest(model, lag = i,
                                              prewhite = FALSE)[2,2] %>% sqrt()
                }
                low <- min(results)+qnorm((1-conf_level)/2)*max(HAC)
                up <- max(results)-qnorm((1-conf_level)/2)*max(HAC)

                plot(results, main = main, type = "l", col = "red",
                     lwd = 1.5, ylab = "Impulse Response", xlab = "Horizon",
                     ylim = c(low, up), axes = FALSE)

                axis(1, at = 1:(h+1), labels = 0:h, cex.axis = 0.75)
                axis(2, at = seq(low, up, (up-low)/4),
                     labels = format(seq(low, up, (up-low)/4),
                                     digits = 1, scientific = FALSE))
                lines(results + qnorm((1-conf_level)/2) * HAC, col = "blue")
                lines(results - qnorm((1-conf_level)/2) * HAC, col = "blue")
        }
}
```

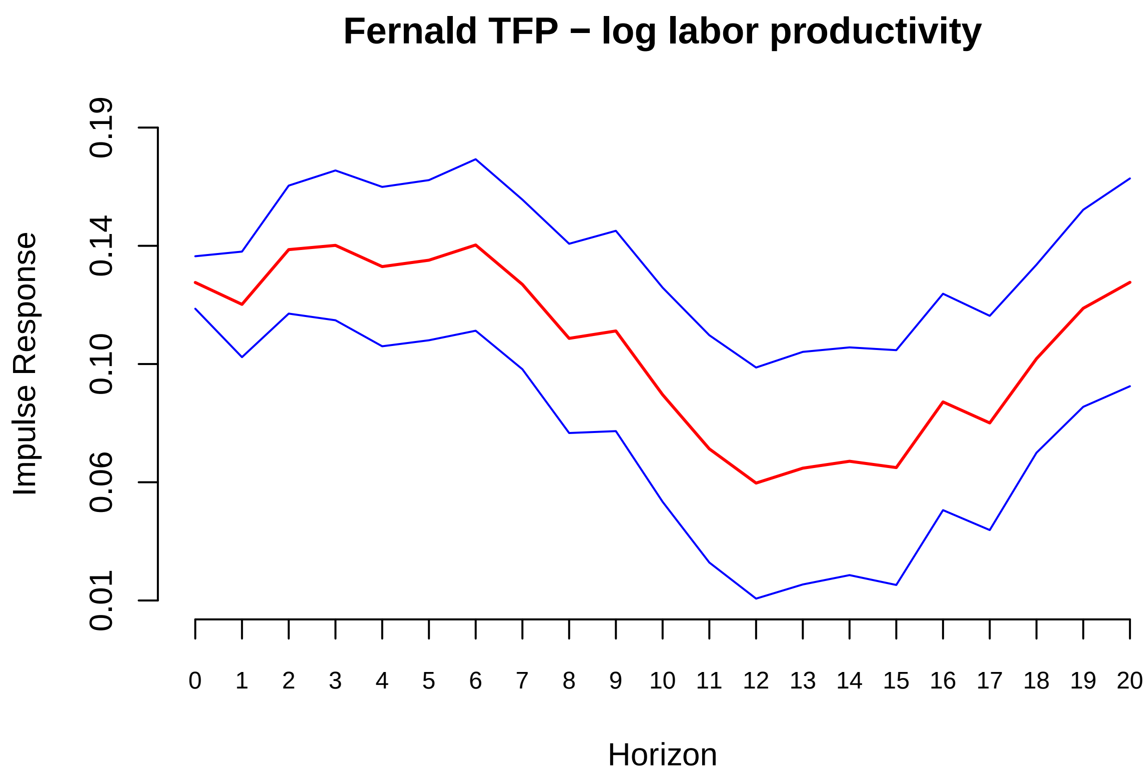
log real GDP per capita



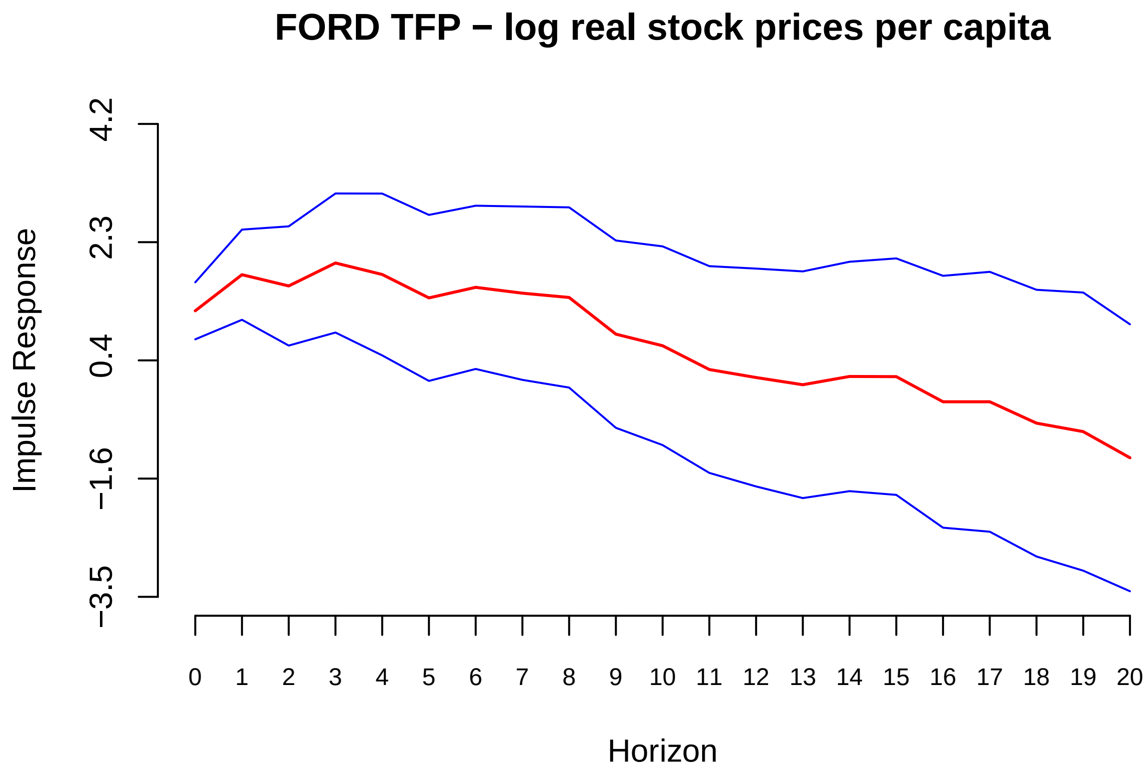


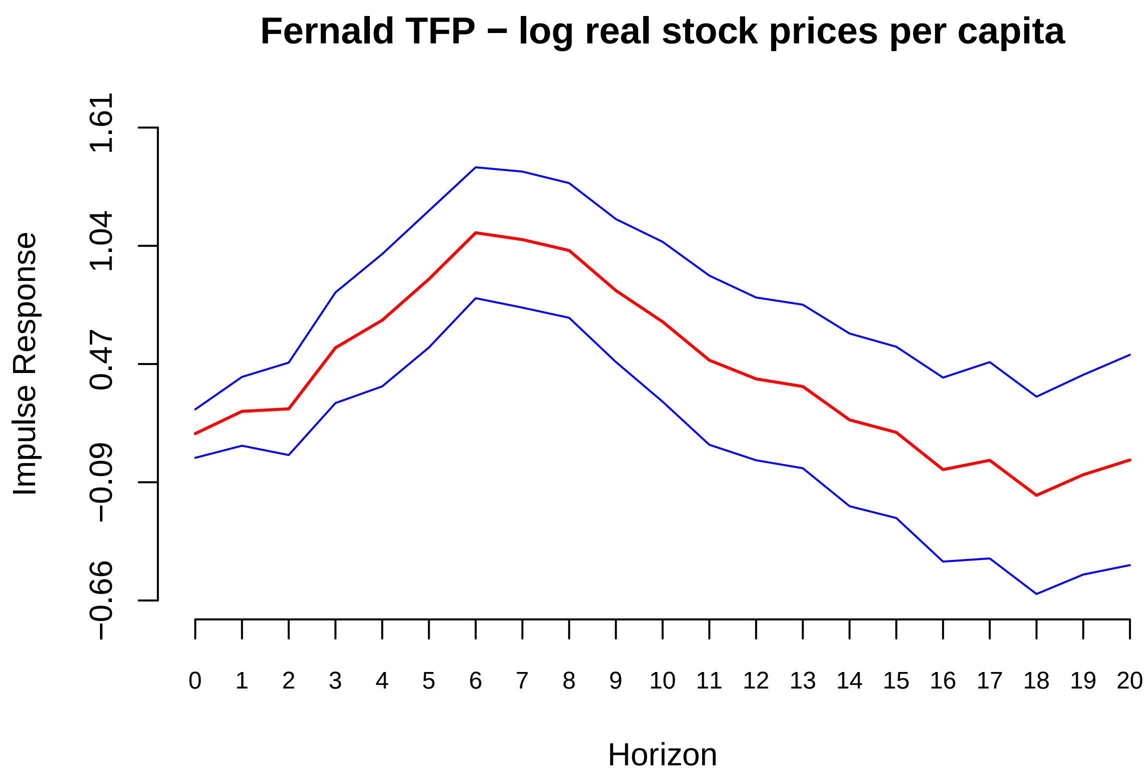
log labor productivity



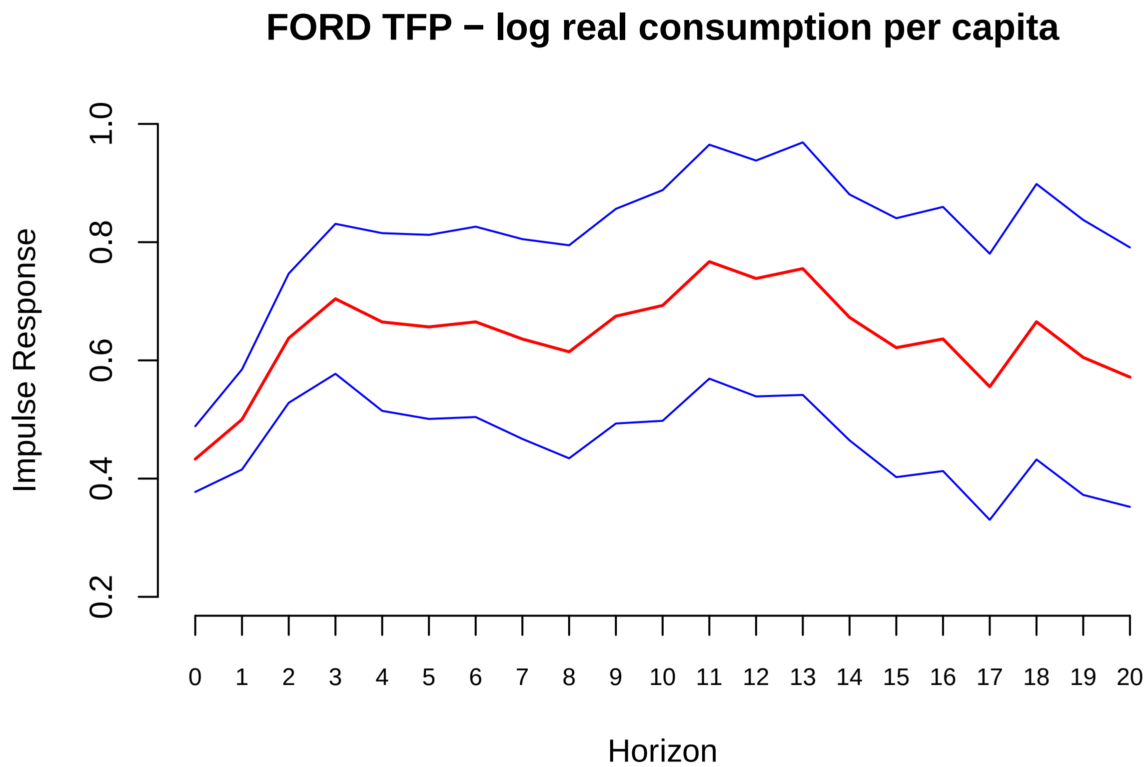


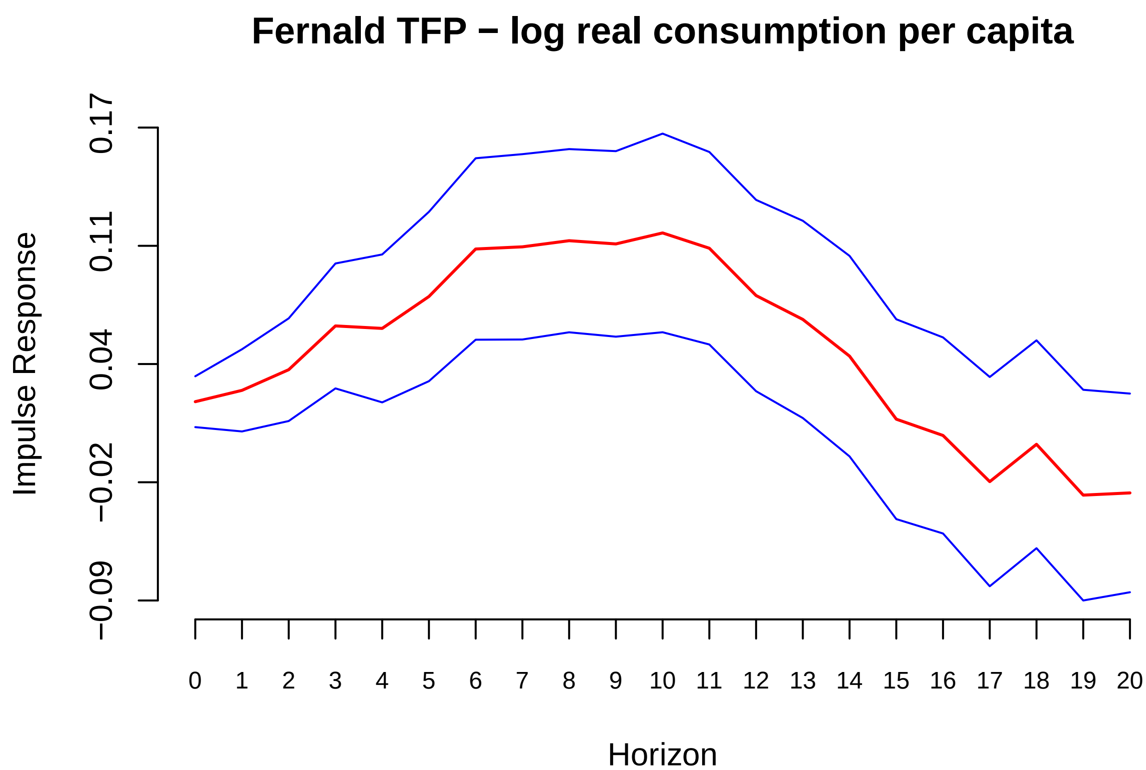
log real stock prices per capita



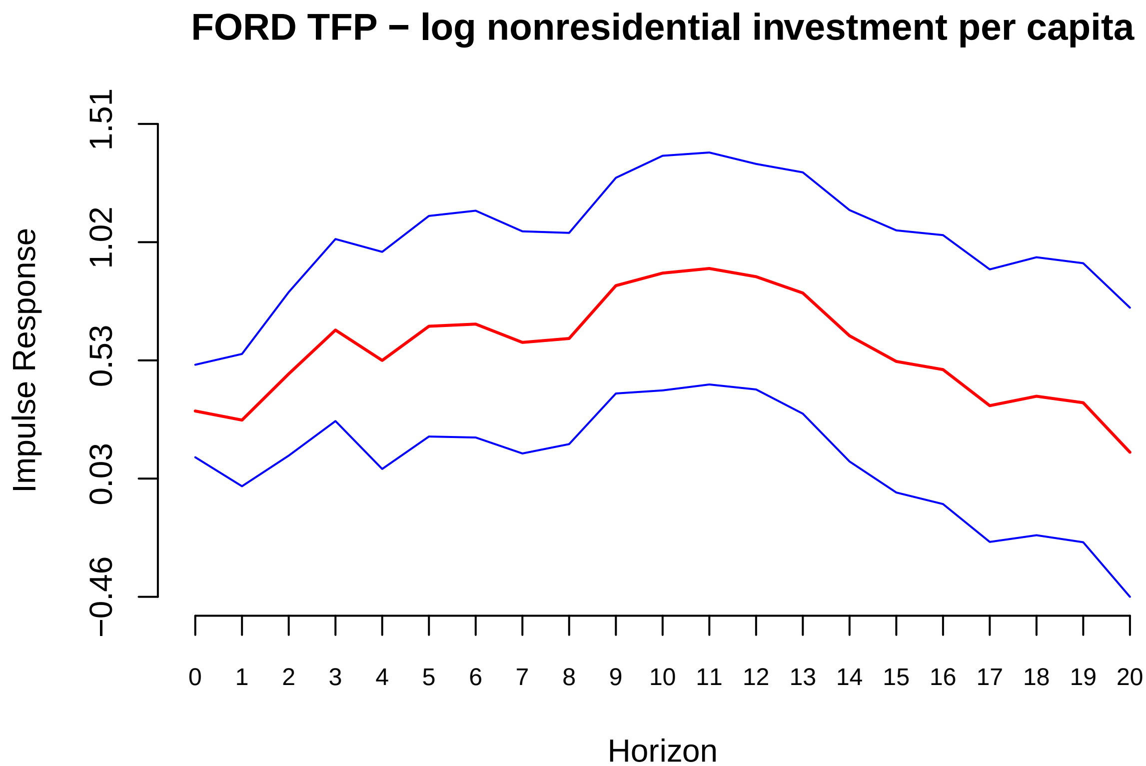


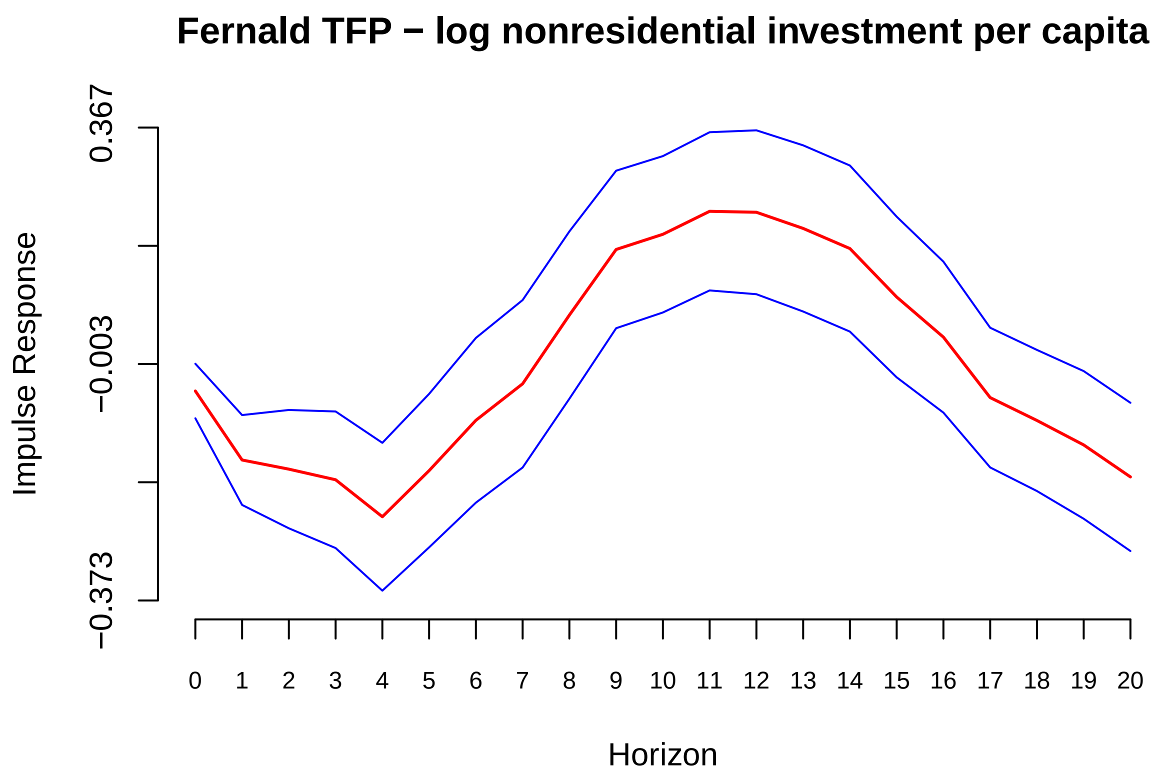
log real consumption per capita





log nonresidential investment per capita





Economic interpretation

The TFP shocks, on average, would stimulate a positive response from real GDP per capita. From the plots for both FORD and Fernald series, we are able to see an upward trending in the percentage change in GDP per capita, yet this positive response grows at a diminishing rate as the horizon stretches – It culminates/ peaks at a horizon of 10-12 quarters or 2.5-3 years and then begins reverting back to zero. This is true for both kinds of TFP shocks, indicating that the stimulating response of real GDP per capita (economic growth) to TFP shocks is not instant but gradual. The confidence bands tell us that this response from the economy is significantly greater than zero.

The percentage change in labor productivity reacts slightly differently against the two TFP shocks – It fluctuates between 0.25% and 0.45% in response to the FORD TFP but between 0.06% and 0.14% in response to the Fernald TFP. Similar behaviors can be viewed on the plots of stock prices per capita, consumption per capita, and nonresidential investment per capita. As such, it is reasonable to hypothesize that TFP shocks would stimulate or expand the economy (measured by real GDP, consumption, and investment per capita). Furthermore, the positive responses from the economy would be stronger for the FORD TFP than the Fernald TFP.