Project 3 — Panel Models (Recreated & Cleaned)

What this report does

1) Load/install packages
2) Load `patent.csv` (or create & save a realistic synthetic dataset if missing)
3) Descriptive plots & summaries
4) Pooled OLS, FE (1-way & 2-way), and RE
5) F-tests & Hausman test with robust SEs

Overview: This project explores how firm-level innovation metrics (R&D spending and patents) affect stock performance over time. Using synthetic data that mimics 100 firms from 1972–1981, I estimated econometric models (Pooled OLS, Fixed Effects, Random Effects) and tested for firm and time heterogeneity.

> Keep `patent.csv` next to this file. If it's missing, a synthetic file is created.

```
needs <- c("plm","AER","ggplot2","corrplot","sandwich","lmtest","dplyr","broom")
to_install <- setdiff(needs, rownames(installed.packages()))
if (length(to_install)) install.packages(to_install, quiet = TRUE)
invisible(lapply(needs, library, character.only = TRUE))
```

Load data (or create if missing)

Dataset Overview: id: Firm identifier (1–100) year: Observed year (1972–1981) rnd.inmllns: Research & development spending (in millions) sales.inmllns: Firm sales (in millions) patents.applied / patents.granted: Innovation output measures return: Stock return percentage stock.price: Simulated firm stock price Note: The dataset is synthetic but structured to mirror realistic relationships between firm innovation and market performance.

```
csv_path <- "patent.csv"
use_synthetic <- FALSE

mk_synth <- function(path = "patent_synthetic.csv") {
  set.seed(42)
  n_firms <- 100
```

```r
  years   <- 72:81
  N       <- n_firms * length(years)

  id <- rep(1:n_firms, each = length(years))
  year <- rep(years, times = n_firms)

  firm_fe <- rnorm(n_firms, 0, 10)[id]
  time_fe <- rnorm(length(years), 0, 5)[match(year, years)]
  industry <- sample(c("Chemicals","Electronics","Food","Pharma","Other"), N, replace = TRUE)

  sales.inmllns <- exp(rnorm(N, 5, 0.6))
  rnd.inmllns   <- pmax(0, 0.08 * sales.inmllns + rnorm(N, 0, 5))
  lrnd          <- log(pmax(rnd.inmllns, 1e-3))
  lsales        <- log(pmax(sales.inmllns, 1e-3))

  patents.applied <- rpois(N, lambda = pmax(1, 0.03*sales.inmllns + 0.5*lrnd))
  patents.granted <- pmax(0, round(patents.applied * runif(N, 0.4, 0.9)))

  `return` <- 2 + 0.04*patents.applied + 0.07*patents.granted + rnorm(N, 0, 2)

  stock.price <- 20 + 0.5*`return` + 0.1*patents.applied + 0.2*patents.granted +
                3*lrnd + 2*lsales + firm_fe + time_fe + rnorm(N, 0, 5)

  df <- data.frame(
    id, year, `return`,
    patents.applied, patents.granted, stock.price,
    rnd.inmllns, sales.inmllns, lrnd, lsales, industry,
    stringsAsFactors = FALSE
  )

  for (yy in 72:81) df[[paste0("y", yy)]] <- as.integer(df$year == yy)

  write.csv(df, path, row.names = FALSE)
  path
}

if (!file.exists(csv_path)) {
  message("INFO: 'patent.csv' not found. Using/creating 'patent_synthetic.csv'.")
  csv_path <- "patent_synthetic.csv"
  use_synthetic <- TRUE
  if (!file.exists(csv_path)) csv_path <- mk_synth(csv_path)
}

patent <- read.csv(csv_path, stringsAsFactors = FALSE)

# Basic checks/coercions
expected_cols <- c("id","year","return","patents.applied","patents.granted","stock.price",
```

```
                    "rnd.inmllns","sales.inmllns","lrnd","lsales","industry")
stopifnot(all(expected_cols %in% names(patent)))

patent$id <- as.factor(patent$id)
patent$year <- as.integer(patent$year)
patent$industry <- as.factor(patent$industry)

for (yy in 72:81) if (!paste0("y",yy) %in% names(patent)) patent[[paste0("y",yy)]] <- as.intege
```

Descriptive analysis & plots

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

# Boxplot of stock price
boxplot(patent$stock.price, main = "Boxplot: Stock Prices (1972-1981)", horizontal = TRUE)
summary(patent$stock.price)
```

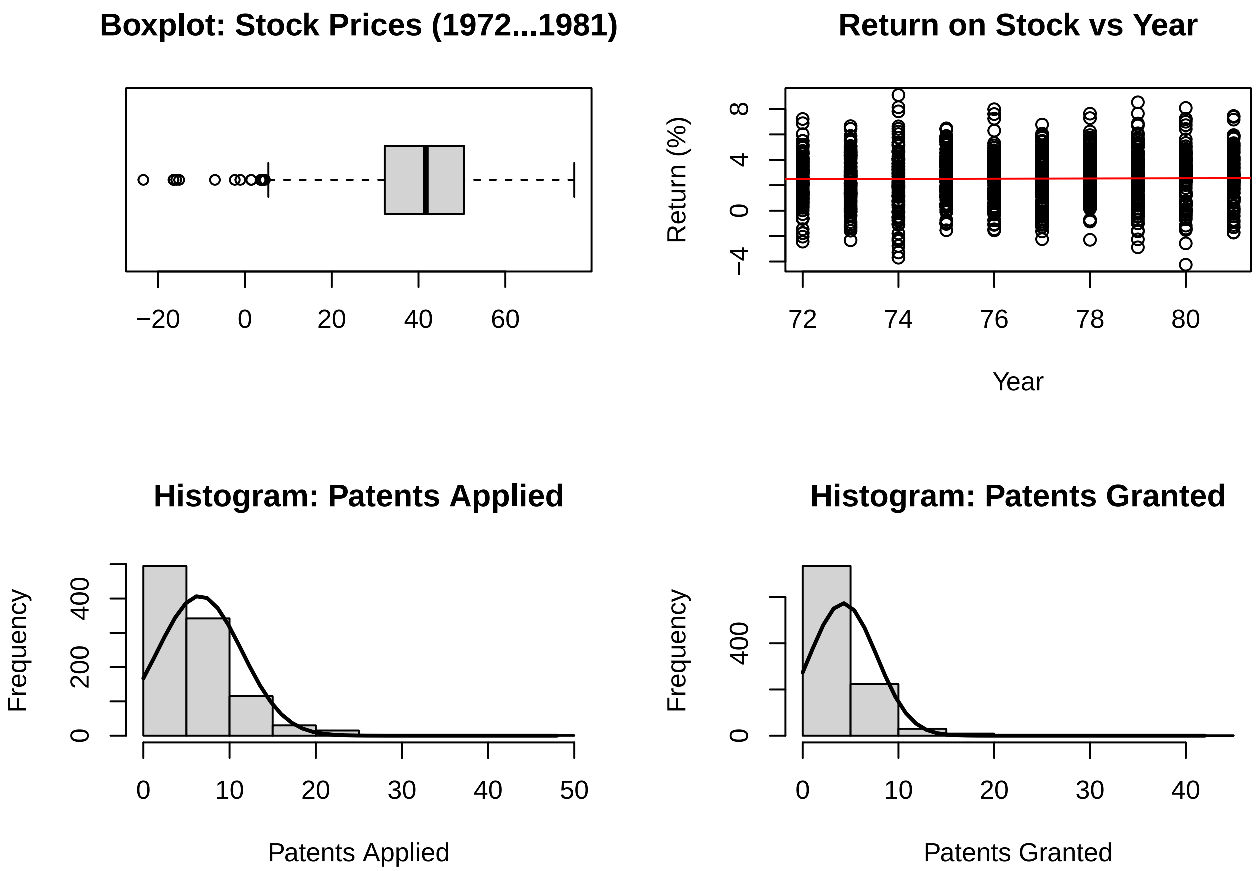
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -23.39   32.21   41.66   40.78   50.51   75.90
```

```
# Scatter: return over year
plot(patent$year, patent$`return`, main = "Return on Stock vs Year", xlab = "Year", ylab = "Re
abline(lm(`return` ~ year, data = patent), col = "red")
summary(patent$`return`)
```

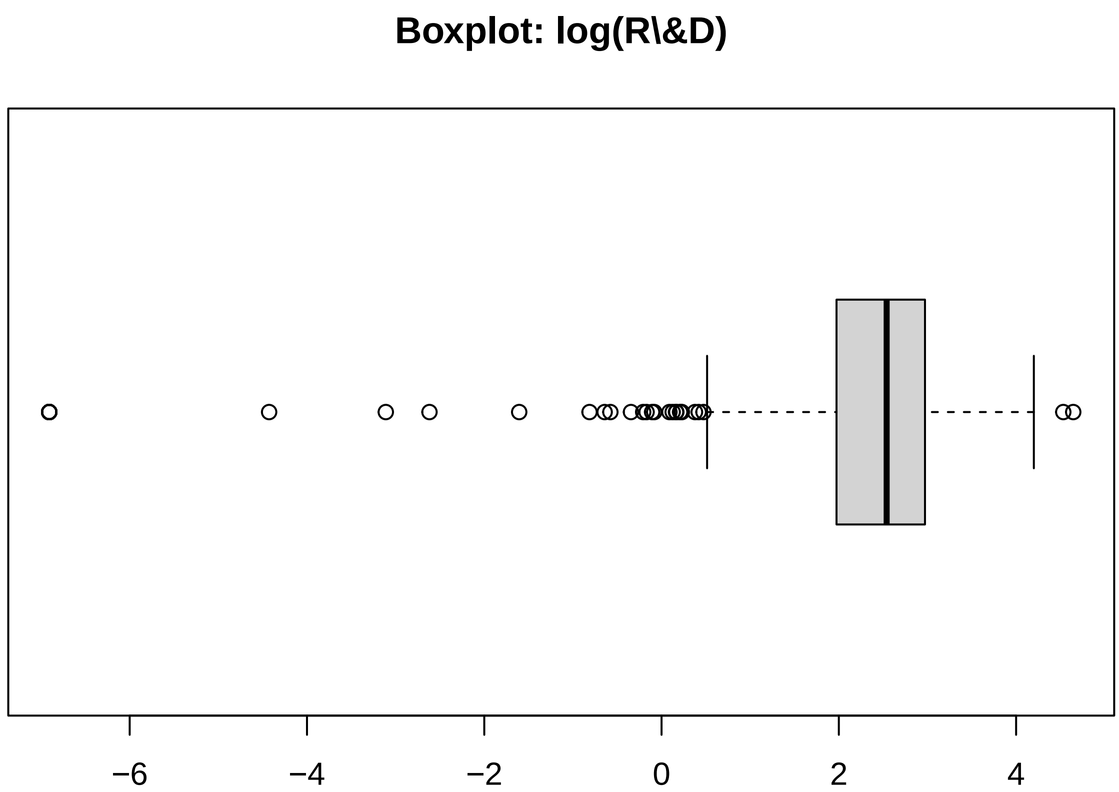
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -4.251   1.089   2.610   2.520   3.951   9.098
```

```
# Hist: patents applied
x <- patent$patents.applied
h <- hist(x, breaks = 10, xlab = "Patents Applied", main = "Histogram: Patents Applied")
xfit <- seq(min(x, na.rm=TRUE), max(x, na.rm=TRUE), length=40)
yfit <- dnorm(xfit, mean=mean(x, na.rm=TRUE), sd=sd(x, na.rm=TRUE))
yfit <- yfit * diff(h$mids[1:2]) * length(na.omit(x)); lines(xfit, yfit, lwd=2)

# Hist: patents granted
xg <- patent$patents.granted
hg <- hist(xg, breaks = 10, xlab = "Patents Granted", main = "Histogram: Patents Granted")
xfitg <- seq(min(xg, na.rm=TRUE), max(xg, na.rm=TRUE), length=40)
yfitg <- dnorm(xfitg, mean=mean(xg, na.rm=TRUE), sd=sd(xg, na.rm=TRUE))
yfitg <- yfitg * diff(hg$mids[1:2]) * length(na.omit(xg)); lines(xfitg, yfitg, lwd=2)
```



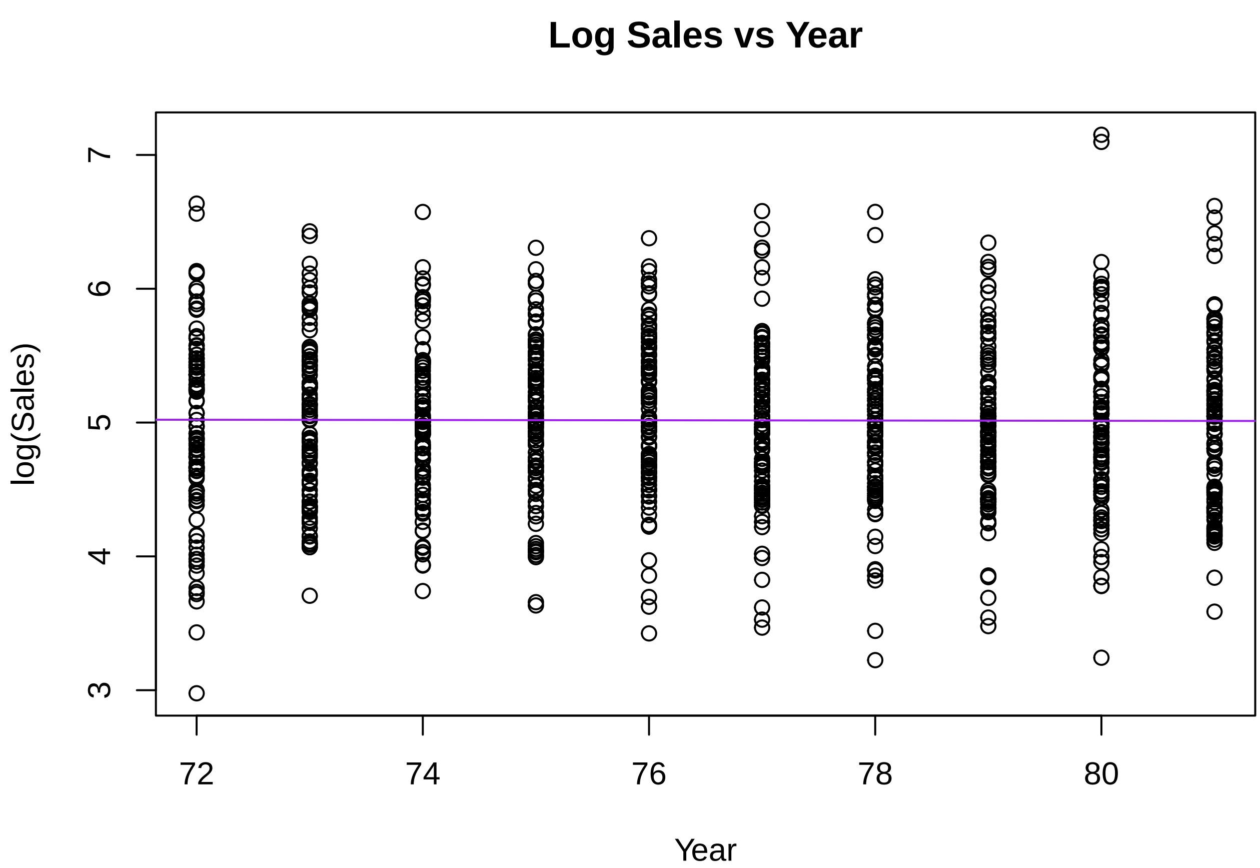

```r
# Reset layout
par(mfrow = c(1,1))
boxplot(patent$lrnd, main = "Boxplot: log(R\\&D)", horizontal = TRUE)
```




```
summary(patent$lrnd)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -6.908   1.974   2.540   2.126   2.972   4.644
```

```
plot(patent$year, patent$lsales, main = "Log Sales vs Year", xlab = "Year", ylab = "log(Sales)"
abline(lm(lsales ~ year, data = patent), col = "purple")
```



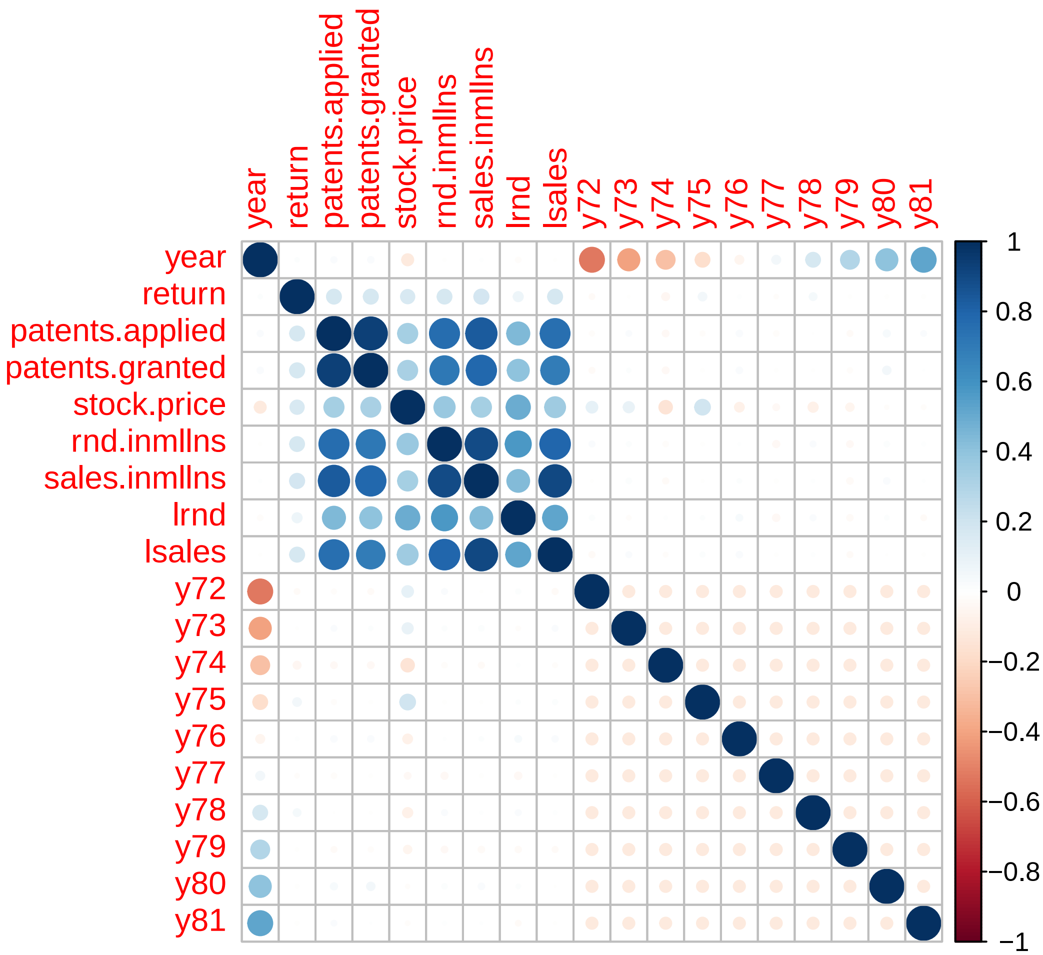

```r
summary(patent$lsales)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.977   4.599   5.010   5.017   5.431   7.151
```

```r
numeric_cols <- sapply(patent, is.numeric)
P <- tryCatch(cor(patent[, numeric_cols], use = "pairwise.complete.obs"), error = function(e)
if (!is.null(P)) {
  print(round(P[1:min(6,nrow(P)), 1:min(6,ncol(P))], 2))
  corrplot(P, method = "circle")
}
```

```
##                    year return patents.applied patents.granted stock.price
## year               1.00   0.01            0.02            0.03       -0.11
## return             0.01   1.00            0.18            0.18        0.16
## patents.applied    0.02   0.18            1.00            0.94        0.33
## patents.granted    0.03   0.18            0.94            1.00        0.33
## stock.price       -0.11   0.16            0.33            0.33        1.00
## rnd.inmllns        0.00   0.17            0.77            0.72        0.37
##                  rnd.inmllns
## year                    0.00
## return                  0.17
```

```
## patents.applied        0.77
## patents.granted        0.72
## stock.price            0.37
## rnd.inmllns            1.00
```



Panel models & heterogeneity

Methodology # I applied several econometric models to estimate how innovation # variables influence stock price: # - Pooled OLS assumes no firm-specific effects. # - Fixed Effects (FE) controls for unobserved time-invariant firm characteristics. # - Random Effects (RE) assumes those effects are random. # I compared these models using F-tests (for fixed effects significance) # and a Hausman test (to decide between FE and RE). # ————————————————————

```
patent.pd <- pdata.frame(patent, index = c("id", "year"))

# Pooled OLS
mreg.pooled <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
```

```r
                   model = "pooling", data = patent.pd)
cat("\n=== Pooled OLS ===\n"); print(summary(mreg.pooled))
```

```
##
## === Pooled OLS ===

## Pooling Model
##
## Call:
## plm(formula = stock.price ~ return + patents.applied + patents.granted +
##     lrnd + lsales, data = patent.pd, model = "pooling")
##
## Balanced Panel: n = 100, T = 10, N = 1000
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -42.59880  -7.61075   0.44801   8.51990  29.89280
##
## Coefficients:
##                  Estimate Std. Error t-value  Pr(>|t|)
## (Intercept)     23.09105    4.57038  5.0523 5.192e-07 ***
## return           0.70819    0.19431  3.6447 0.0002816 ***
## patents.applied -0.17942    0.25164 -0.7130 0.4760092
## patents.granted  0.64334    0.32133  2.0021 0.0455437 *
## lrnd             3.18099    0.24286 13.0980 < 2.2e-16 ***
## lsales           1.51339    1.04790  1.4442 0.1489922
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    212550
## Residual Sum of Squares: 154260
## R-Squared:      0.27425
## Adj. R-Squared: 0.2706
## F-statistic: 75.1225 on 5 and 994 DF, p-value: < 2.22e-16
```

```r
# Cluster-robust SEs
pool_robust <- vcovHC(mreg.pooled, type="HC0", cluster="group")
cat("\n=== Pooled OLS (cluster-robust SEs) ===\n"); print(coeftest(mreg.pooled, vcov = pool_rol
```

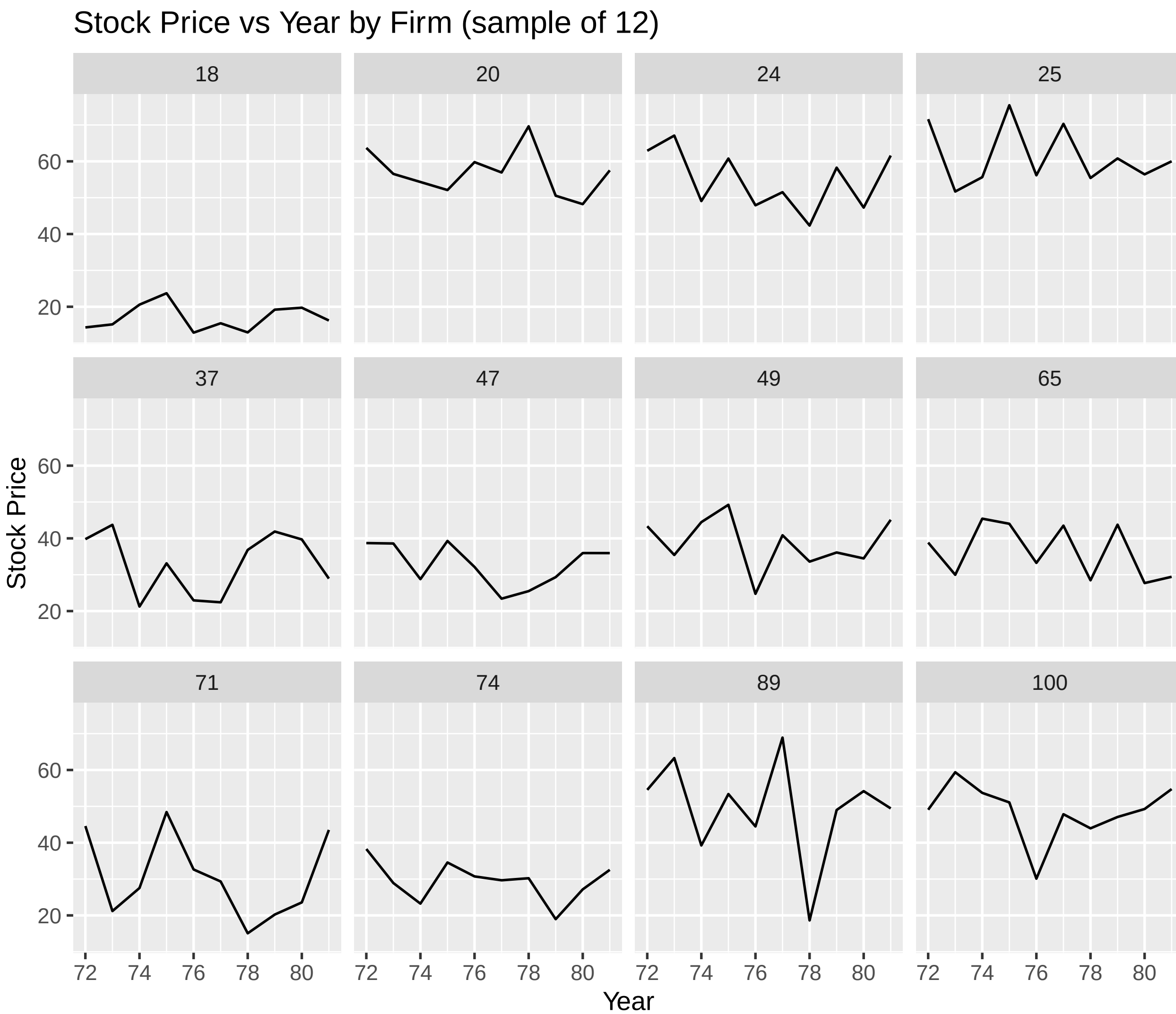
```
##
## === Pooled OLS (cluster-robust SEs) ===

##
## t test of coefficients:
##
##                  Estimate Std. Error t value  Pr(>|t|)
```

```
## (Intercept)     23.09105    3.62345  6.3727 2.838e-10 ***
## return           0.70819    0.20602  3.4375 0.0006114 ***
## patents.applied -0.17942    0.19462 -0.9219 0.3568070
## patents.granted  0.64334    0.26434  2.4338 0.0151182 *
## lrnd             3.18099    0.29535 10.7701 < 2.2e-16 ***
## lsales           1.51339    0.87180  1.7359 0.0828860 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

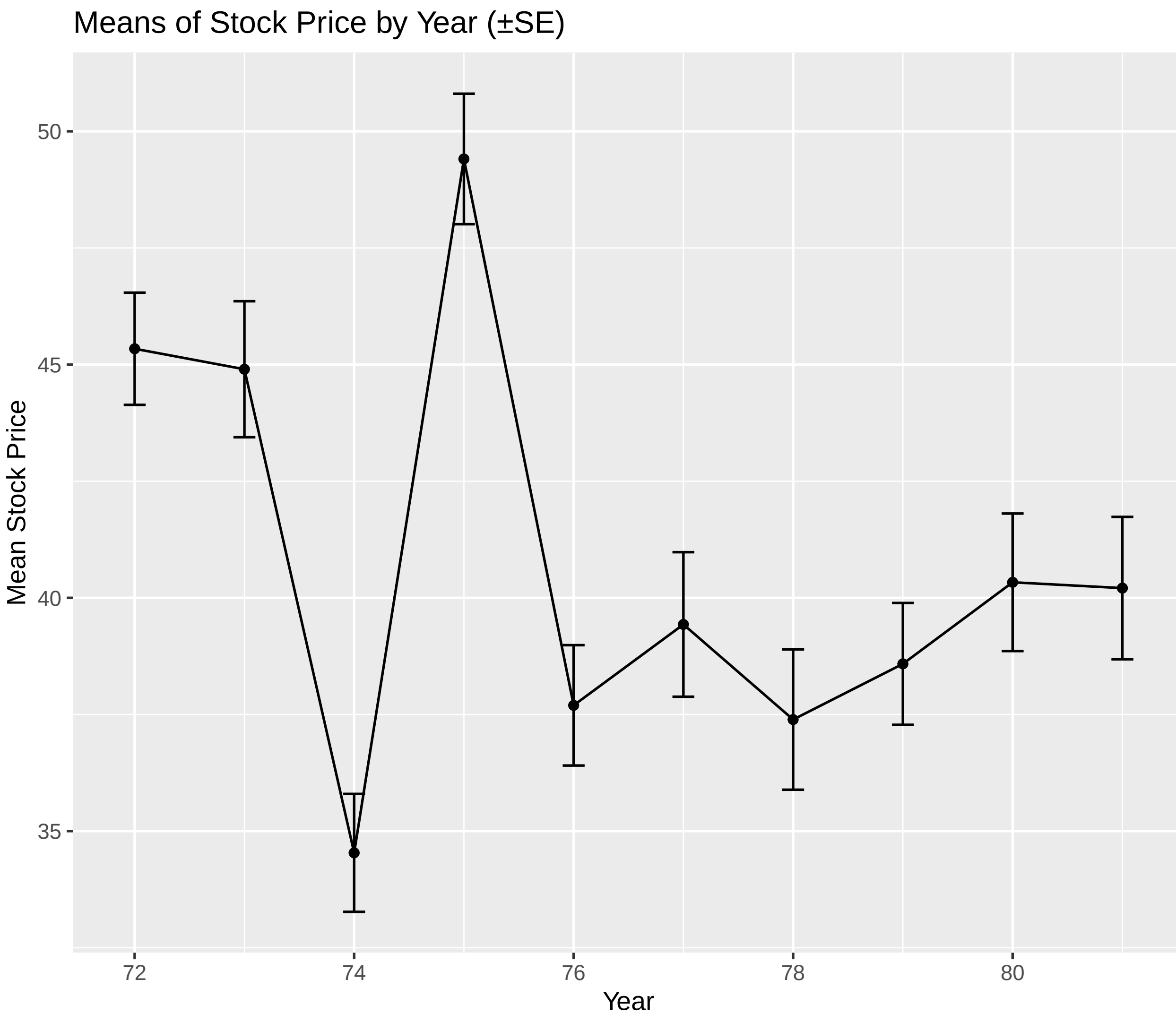
```r
# Heterogeneity visuals (ggplot2)
set.seed(42)
ids_sample <- sample(unique(patent$id), 12)
pat_sub <- subset(patent, id %in% ids_sample)

# Faceted lines for a sample of firms
ggplot(pat_sub, aes(year, stock.price)) +
  geom_line() +
  facet_wrap(~ id, ncol = 4) +
  labs(title = "Stock Price vs Year by Firm (sample of 12)",
       x = "Year", y = "Stock Price")
```



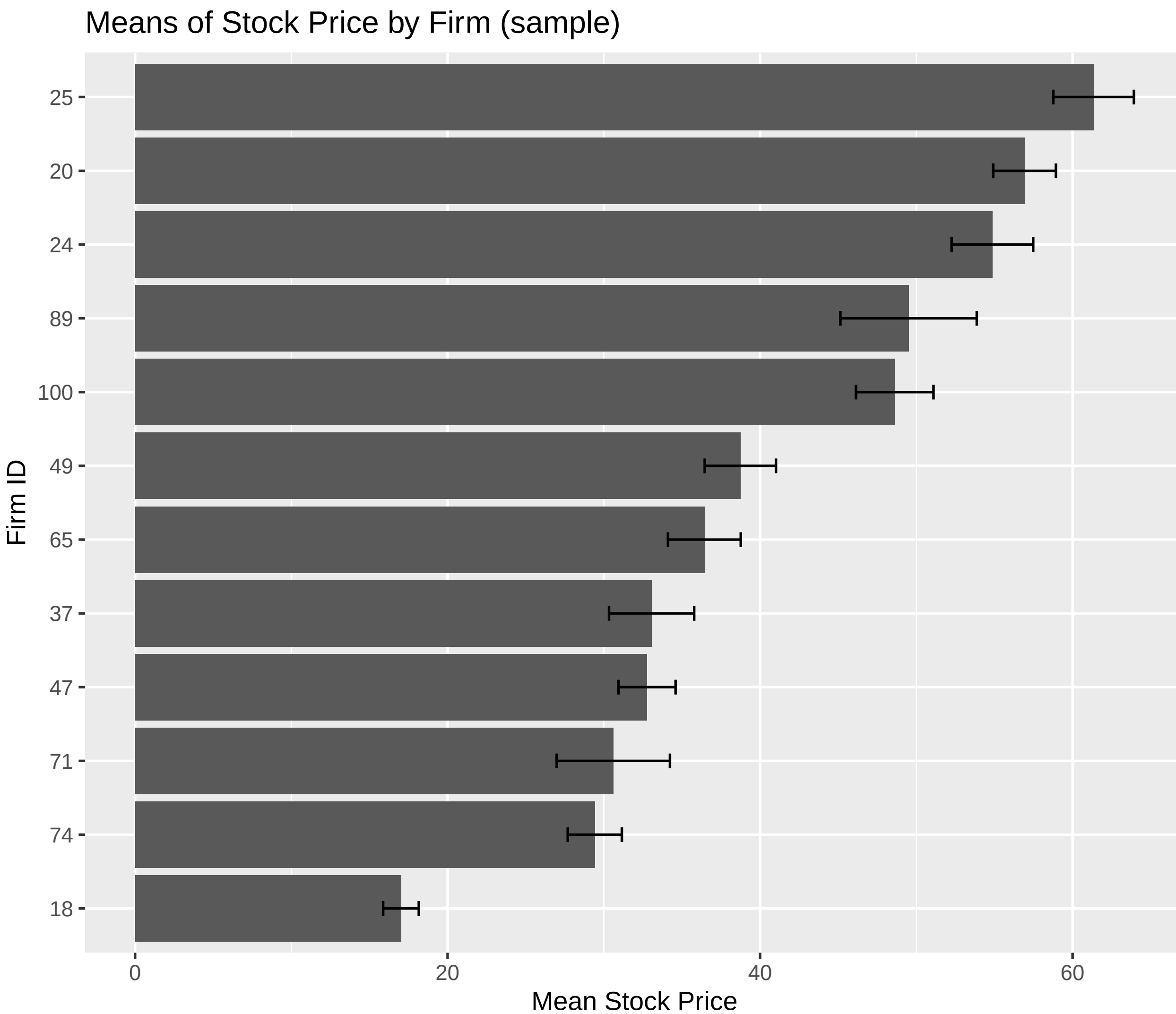

```r
# Means by year with SE
by_year <- patent |>
  dplyr::group_by(year) |>
  dplyr::summarise(mean_sp = mean(stock.price, na.rm=TRUE),
                   se_sp = sd(stock.price, na.rm=TRUE)/sqrt(dplyr::n()),
                   .groups = "drop")

ggplot(by_year, aes(year, mean_sp)) +
  geom_line() + geom_point() +
  geom_errorbar(aes(ymin = mean_sp - se_sp, ymax = mean_sp + se_sp), width = 0.2) +
  labs(title = "Means of Stock Price by Year (±SE)", x = "Year", y = "Mean Stock Price")
```




```r
# Means by firm (sample) with SE
by_firm <- pat_sub |>
  dplyr::group_by(id) |>
  dplyr::summarise(mean_sp = mean(stock.price, na.rm=TRUE),
                   se_sp = sd(stock.price, na.rm=TRUE)/sqrt(dplyr::n()),
                   .groups = "drop")

ggplot(by_firm, aes(x = reorder(id, mean_sp), y = mean_sp)) +
  geom_col() +
  geom_errorbar(aes(ymin = mean_sp - se_sp, ymax = mean_sp + se_sp), width = 0.2) +
  coord_flip() +
  labs(title = "Means of Stock Price by Firm (sample)", x = "Firm ID", y = "Mean Stock Price")
```



Fixed Effects, F-tests, and Two-Way FE

```
mreg.within <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
                   model = "within", data = patent.pd, effect = "individual")
cat("\n=== Fixed Effects (Within) ===\n"); print(summary(mreg.within))
```

```
##
## === Fixed Effects (Within) ===

## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = stock.price ~ return + patents.applied + patents.granted +
##     lrnd + lsales, data = patent.pd, effect = "individual", model = "within")
##
```

```
## Balanced Panel: n = 100, T = 10, N = 1000
##
## Residuals:
##     Min.  1st Qu.   Median  3rd Qu.     Max.
## -17.0651  -4.6225  -0.1589   4.3140  20.4984
##
## Coefficients:
##                  Estimate Std. Error t-value  Pr(>|t|)
## return           0.714844   0.112905  6.3313  3.84e-10 ***
## patents.applied -0.068493   0.143672 -0.4767  0.633668
## patents.granted  0.456623   0.184611  2.4734  0.013567 *
## lrnd             3.030403   0.141572 21.4054 < 2.2e-16 ***
## lsales           1.625603   0.600888  2.7053  0.006953 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    89240
## Residual Sum of Squares: 41718
## R-Squared:      0.53252
## Adj. R-Squared: 0.4782
## F-statistic: 203.905 on 5 and 895 DF, p-value: < 2.22e-16
```

```
cat("\n=== Pooled vs FE (F-test) ===\n"); print(pFtest(mreg.within, mreg.pooled))
```

```
##
## === Pooled vs FE (F-test) ===
```

```
##
##  F test for individual effects
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## F = 24.388, df1 = 99, df2 = 895, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

```
fm_full <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
               data = patent.pd, model = "within", effect = "twoways")
fm_time <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
               data = patent.pd, model = "within", effect = "time")
fm_firm <- mreg.within
fm_no   <- mreg.pooled

cat("\n=== F-tests vs pooled ===\n")
```

```
##
## === F-tests vs pooled ===
```

```r
print(pFtest(fm_full, fm_no))
```

```
##
##  F test for twoways effects
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## F = 44.232, df1 = 108, df2 = 886, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

```r
print(pFtest(fm_time, fm_no))
```

```
##
##  F test for time effects
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## F = 14.102, df1 = 9, df2 = 985, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

```r
print(pFtest(fm_firm, fm_no))
```

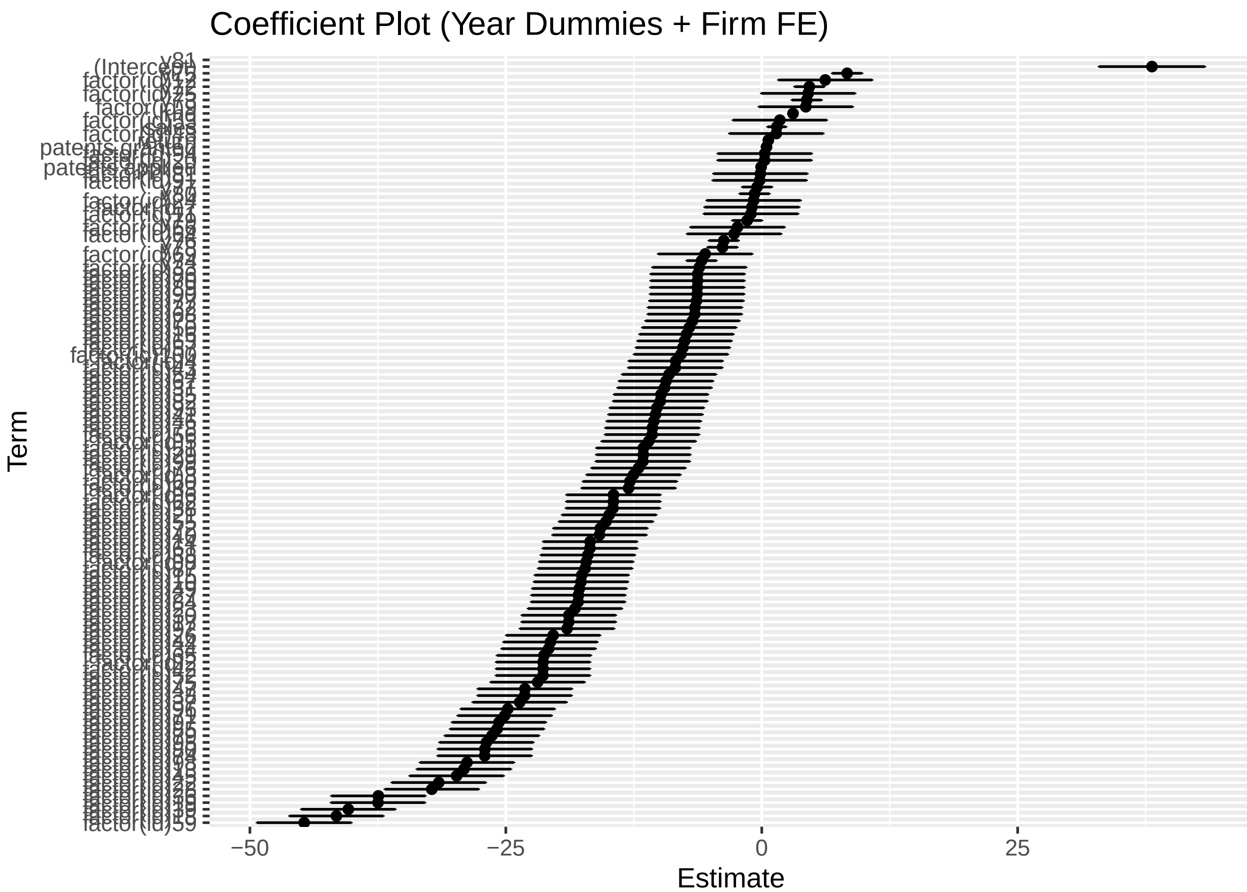
```
##
##  F test for individual effects
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## F = 24.388, df1 = 99, df2 = 895, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

Coefficient plot (ggplot2/broom)

```r
FE_Full_lm <- lm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales +
                   y72 + y73 + y74 + y75 + y76 + y77 + y78 + y79 + y80 + y81 + factor(id),
                 data = patent)

coef_data <- broom::tidy(FE_Full_lm, conf.int = TRUE)

ggplot(coef_data, aes(x = reorder(term, estimate), y = estimate)) +
  geom_point() +
  geom_errorbar(aes(ymin = conf.low, ymax = conf.high), width = 0.2) +
  coord_flip() +
  labs(title = "Coefficient Plot (Year Dummies + Firm FE)",
       x = "Term", y = "Estimate")
```



Random Effects & Hausman test

```
mreg.random <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
                   model = "random", data = patent.pd)
cat("\n=== Random Effects ===\n"); print(summary(mreg.random))
```

```
##
## === Random Effects ===

## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = stock.price ~ return + patents.applied + patents.granted +
##     lrnd + lsales, data = patent.pd, model = "random")
##
## Balanced Panel: n = 100, T = 10, N = 1000
##
## Effects:
##                     var std.dev share
```

```
## idiosyncratic  46.612   6.827  0.29
## individual    113.895  10.672  0.71
## theta: 0.8017
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -19.944328  -4.763477   0.029682   4.175645  21.324288
##
## Coefficients:
##                 Estimate Std. Error z-value  Pr(>|z|)
## (Intercept)     22.90646    2.81678  8.1321 4.218e-16 ***
## return           0.71460    0.11241  6.3574 2.052e-10 ***
## patents.applied -0.07323    0.14314 -0.5116  0.608943
## patents.granted  0.46465    0.18388  2.5269  0.011508 *
## lrnd             3.03711    0.14092 21.5515 < 2.2e-16 ***
## lsales           1.62044    0.59857  2.7072  0.006786 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    94089
## Residual Sum of Squares: 46148
## R-Squared:      0.50953
## Adj. R-Squared: 0.50706
## Chisq: 1032.62 on 5 DF, p-value: < 2.22e-16
```

```r
cat("\n=== Hausman Test (FE vs RE) ===\n")
```

```
##
## === Hausman Test (FE vs RE) ===
```

```r
ht <- try(phtest(mreg.within, mreg.random), silent = TRUE)
if (inherits(ht, "try-error")) {
  cat("Standard Hausman failed; showing coefficient differences on common terms:\n")
  common <- intersect(names(coef(mreg.within)), names(coef(mreg.random)))
  print(coef(mreg.within)[common] - coef(mreg.random)[common])
} else {
  print(ht)
}
```

```
##
##  Hausman Test
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## chisq = 0.55116, df = 5, p-value = 0.9901
## alternative hypothesis: one model is inconsistent
```

Final note

```
##
## NOTE: Using synthetic data; numerical results will differ from the original project, but the
```

Generate firm heterogeneity & time shocks

```r
set.seed(42)

n_firms <- 100            # number of firms
years   <- 72:81          # years (like 1972-1981)
N       <- n_firms * length(years)

id   <- rep(1:n_firms, each = length(years))
year <- rep(years, times = n_firms)

  firm_fe <- rnorm(n_firms, 0, 10)[id]
  time_fe <- rnorm(length(years), 0, 5)[match(year, years)]
  industry <- sample(c("Chemicals","Electronics","Food","Pharma","Other"), N, replace = TRUE)

  sales.inmllns <- exp(rnorm(N, 5, 0.6))                  # ~ log-normal sales
  rnd.inmllns   <- pmax(0, 0.08 * sales.inmllns + rnorm(N, 0, 5))
  lrnd          <- log(pmax(rnd.inmllns, 1e-3))
  lsales        <- log(pmax(sales.inmllns, 1e-3))

  patents.applied <- rpois(N, lambda = pmax(1, 0.03*sales.inmllns + 0.5*lrnd))
  patents.granted <- pmax(0, round(patents.applied * runif(N, 0.4, 0.9)))
```

Return (%) loosely linked to innovation intensity

```r
  `return` <- 2 + 0.04*patents.applied + 0.07*patents.granted + rnorm(N, 0, 2)
```

Stock price with FE + covariates

```r
  stock.price <- 20 + 0.5*`return` + 0.1*patents.applied + 0.2*patents.granted +
    3*lrnd + 2*lsales + firm_fe + time_fe + rnorm(N, 0, 5)

  df <- data.frame(
    id = id, year = year, `return` = `return`,
    patents.applied = patents.applied, patents.granted = patents.granted,
    stock.price = stock.price, rnd.inmllns = rnd.inmllns, sales.inmllns = sales.inmllns,
    lrnd = lrnd, lsales = lsales, industry = industry,
    stringsAsFactors = FALSE
  )
```

Year dummies y72…y81

```r
path <- "patent_synthetic.csv"
write.csv(df, path, row.names = FALSE)

  for (yy in 72:81) df[[paste0("y", yy)]] <- as.integer(df$year == yy)

  write.csv(df, path, row.names = FALSE)
  path
```

```
## [1] "patent_synthetic.csv"
```

```r
if (!file.exists(csv_path)) {
  message("INFO: 'patent.csv' not found. Looking for 'patent_synthetic.csv' or creating one.")
  csv_path <- "patent_synthetic.csv"
  use_synthetic <- TRUE
  if (!file.exists(csv_path)) csv_path <- mk_synth(csv_path)
}

patent <- tryCatch(read.csv(csv_path, stringsAsFactors = FALSE), error = function(e) NULL)
if (!is.data.frame(patent)) stop("Could not load data. Ensure 'patent.csv' or 'patent_syntheti
```

Ensure expected columns exist (minimal checks)

```r
expected_cols <- c("id","year","return","patents.applied","patents.granted","stock.price",
                   "rnd.inmllns","sales.inmllns","lrnd","lsales","industry")
missing_cols <- setdiff(expected_cols, names(patent))
if (length(missing_cols) > 0) stop(paste("Missing expected columns:", paste(missing_cols, colla
```

Coerce & clean types

```r
patent$id <- as.factor(patent$id)
patent$year <- as.integer(patent$year)
patent$industry <- as.factor(patent$industry)
```

Ensure year dummies exist (y72 ... y81)

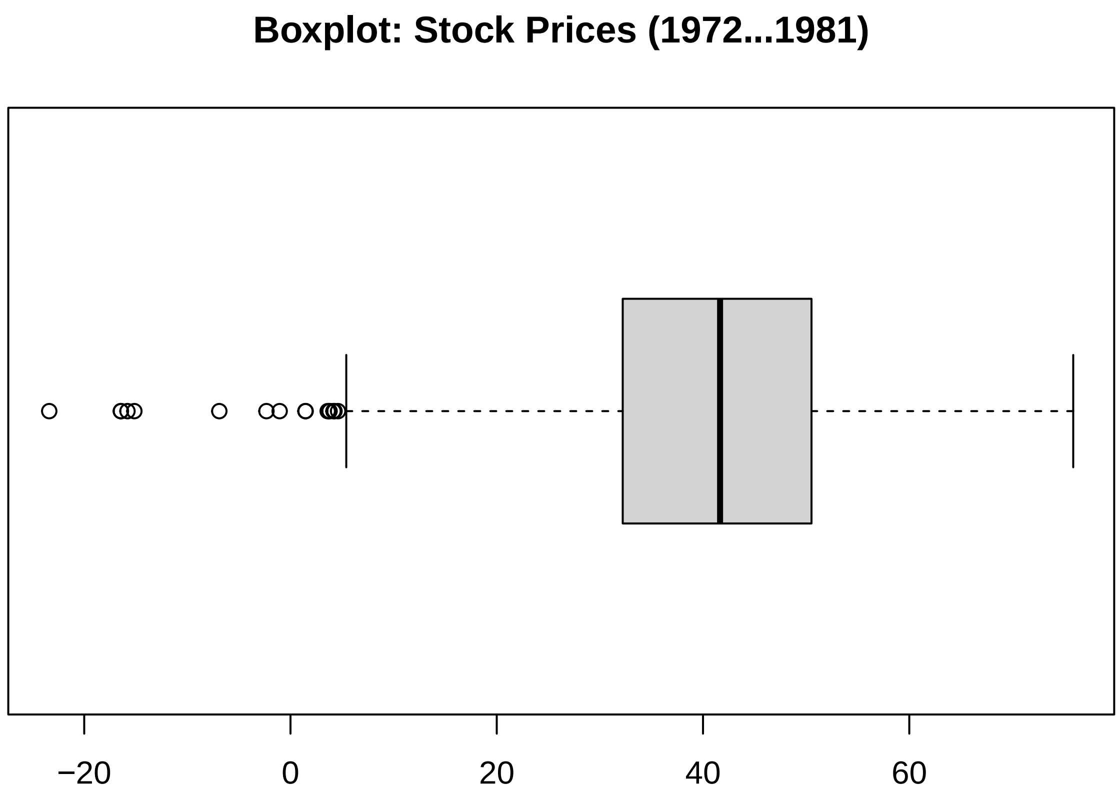
```r
for (yy in 72:81) {
  coln <- paste0("y", yy)
  if (!coln %in% names(patent)) patent[[coln]] <- as.integer(patent$year == yy)
}
```

2) Descriptive analysis & plots

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

Boxplot of stock price

```r
boxplot(patent$stock.price, main="Boxplot: Stock Prices (1972-1981)", horizontal=TRUE)
```

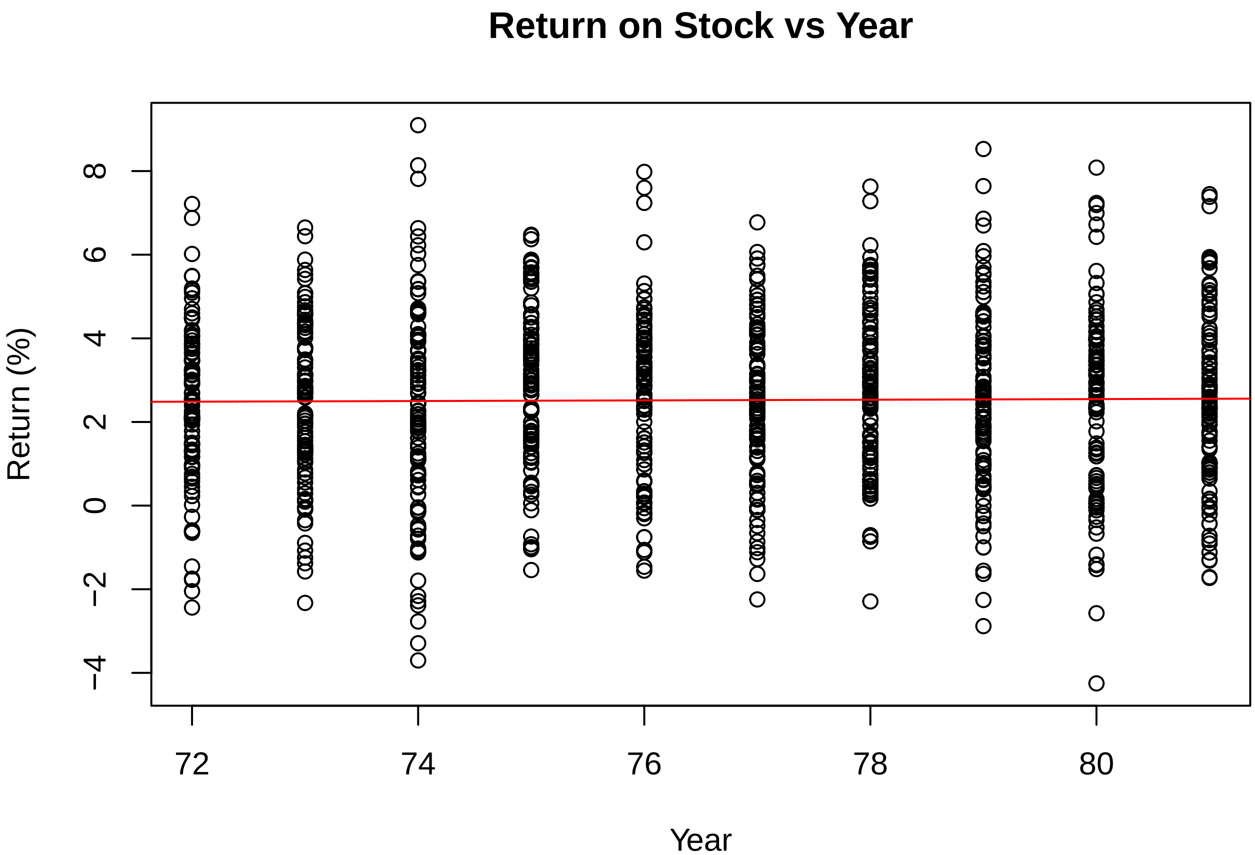



```r
print(summary(patent$stock.price))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -23.39   32.21   41.66   40.78   50.51   75.90
```

Scatter: return over year

```r
plot(patent$year, patent$`return`, main="Return on Stock vs Year", xlab="Year", ylab="Return (%
abline(lm(`return` ~ year, data=patent), col="red")
```

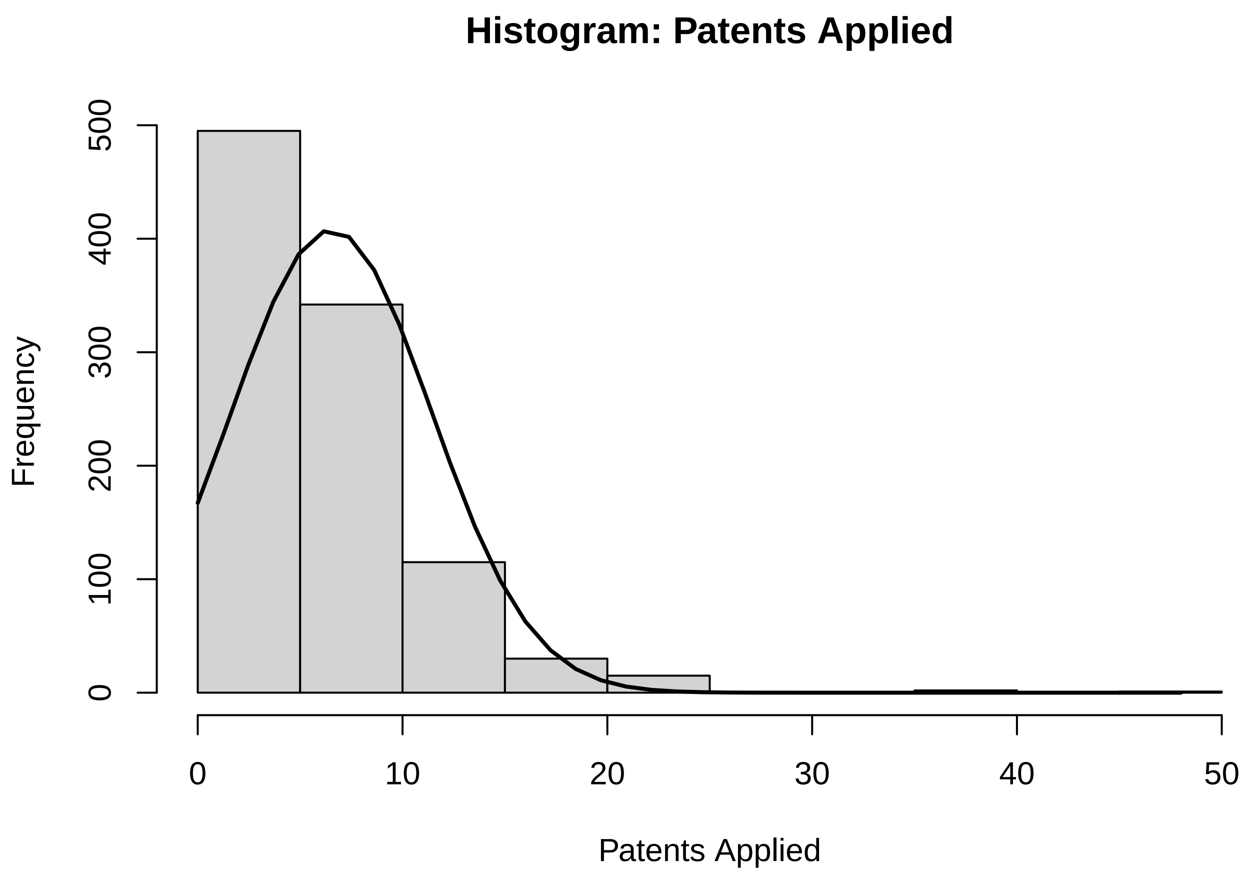



```r
print(summary(patent$`return`))
```

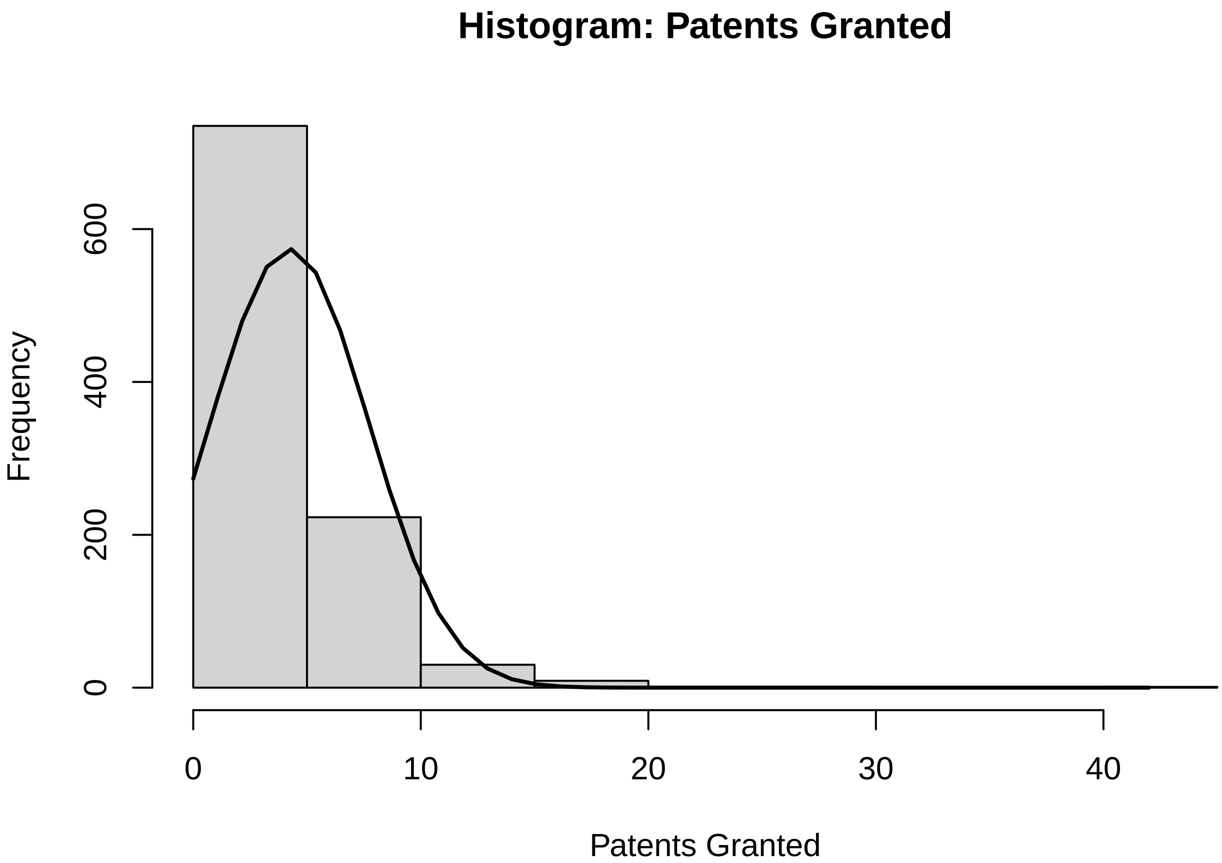
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -4.251   1.089   2.610   2.520   3.951   9.098
```

Histograms: patents applied & granted

```r
x <- patent$patents.applied; h <- hist(x, breaks=10, xlab="Patents Applied", main="Histogram: P
xfit <- seq(min(x, na.rm=TRUE), max(x, na.rm=TRUE), length=40)
yfit <- dnorm(xfit, mean=mean(x, na.rm=TRUE), sd=sd(x, na.rm=TRUE))
yfit <- yfit * diff(h$mids[1:2]) * length(na.omit(x)); lines(xfit, yfit, lwd=2)
```

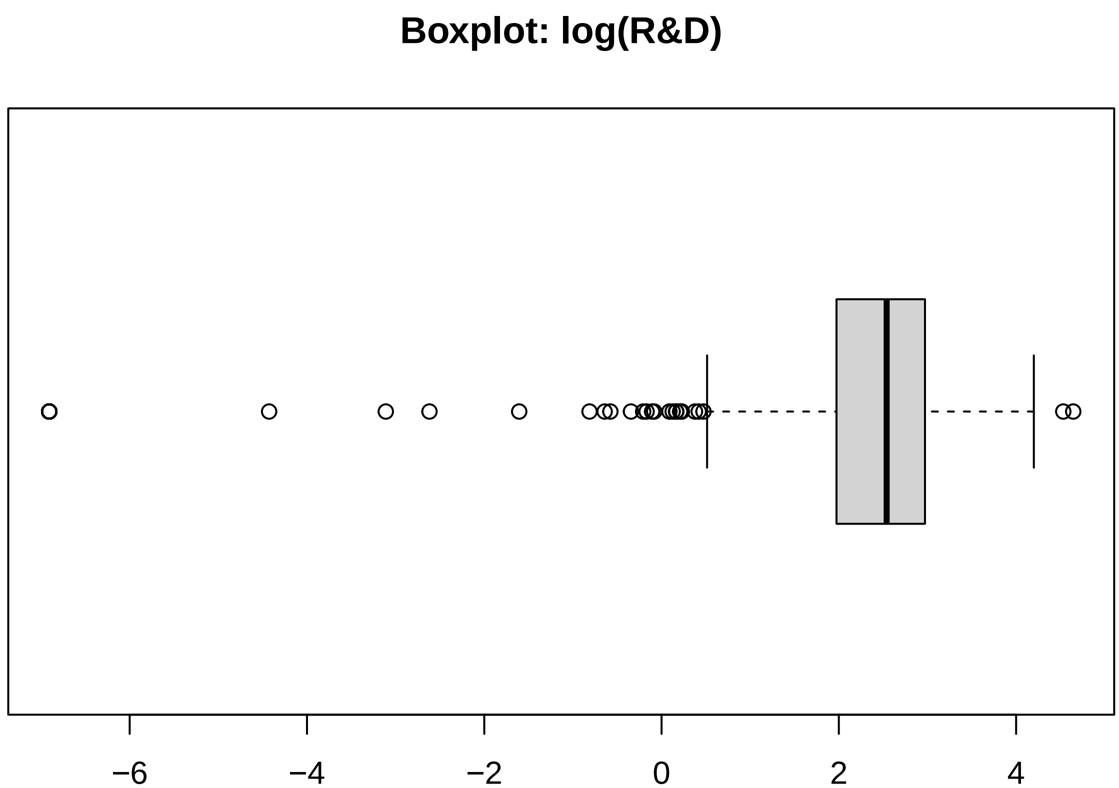



```
xg <- patent$patents.granted; hg <- hist(xg, breaks=10, xlab="Patents Granted", main="Histogram
xfitg <- seq(min(xg, na.rm=TRUE), max(xg, na.rm=TRUE), length=40)
yfitg <- dnorm(xfitg, mean=mean(xg, na.rm=TRUE), sd=sd(xg, na.rm=TRUE))
yfitg <- yfitg * diff(hg$mids[1:2]) * length(na.omit(xg)); lines(xfitg, yfitg, lwd=2)
```



Boxplot: lrnd

```
par(mfrow = c(1,1))
boxplot(patent$lrnd, main="Boxplot: log(R&D)", horizontal=TRUE)
```

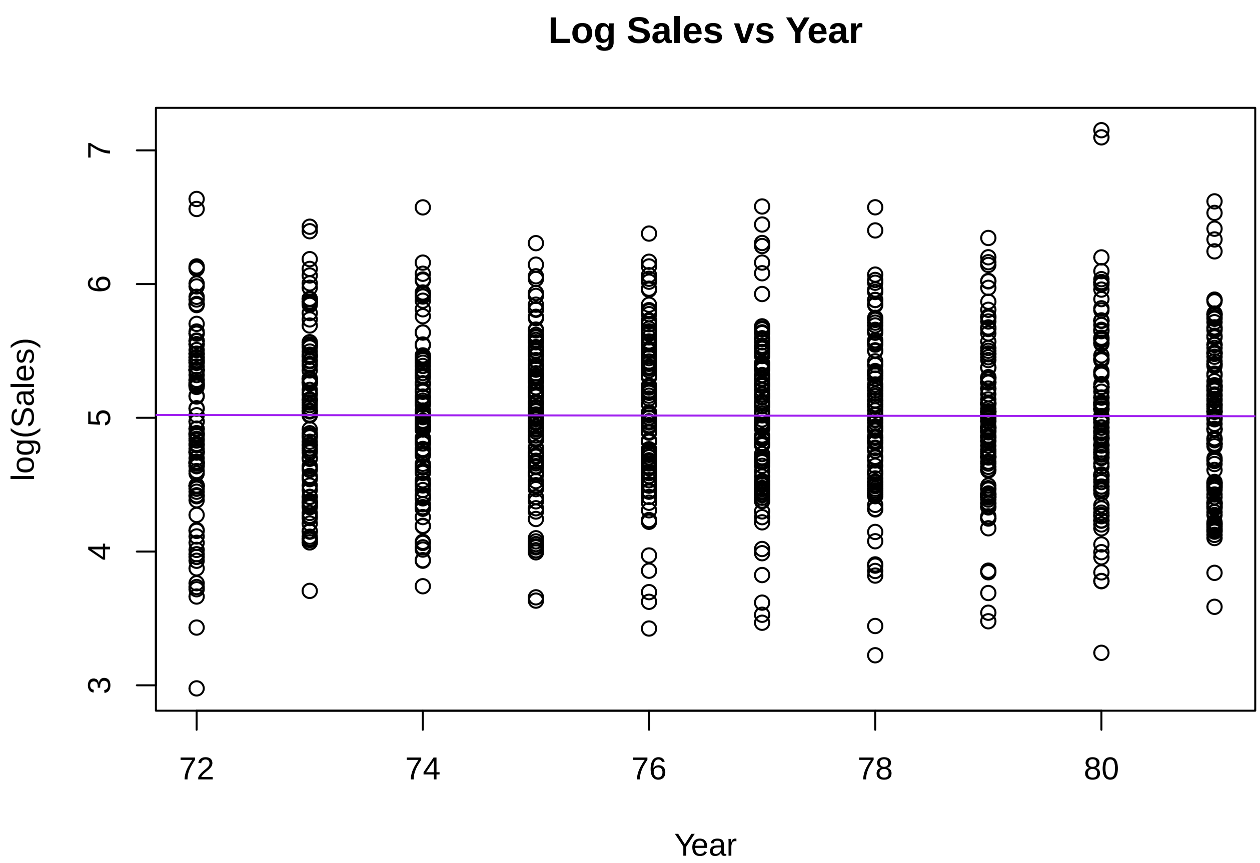



```r
print(summary(patent$lrnd))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -6.908   1.974   2.540   2.126   2.972   4.644
```

Scatter: lsales vs year

```r
plot(patent$year, patent$lsales, main="Log Sales vs Year", xlab="Year", ylab="log(Sales)")
abline(lm(lsales ~ year, data=patent), col="purple")
```




```
print(summary(patent$lsales))
```

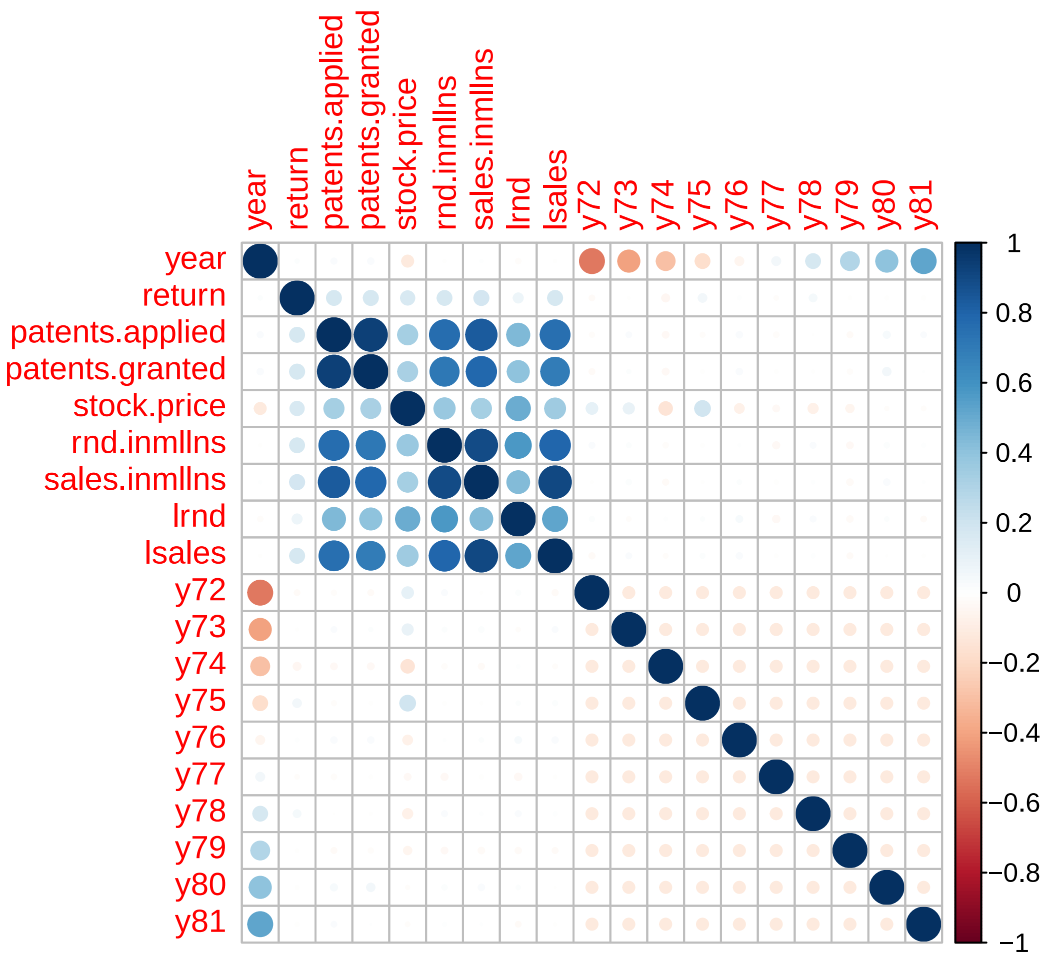
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.977   4.599   5.010   5.017   5.431   7.151
```

Correlation plot (numeric-only)

```
numeric_cols <- sapply(patent, is.numeric)
P <- tryCatch(cor(patent[, numeric_cols], use="pairwise.complete.obs"), error = function(e) NUl
if (!is.null(P)) {
  print(round(P[1:min(6,nrow(P)), 1:min(6,ncol(P))], 2))
  corrplot(P, method="circle")
}
```

```
##                  year return patents.applied patents.granted stock.price
## year             1.00   0.01            0.02            0.03       -0.11
## return           0.01   1.00            0.18            0.18        0.16
## patents.applied  0.02   0.18            1.00            0.94        0.33
## patents.granted  0.03   0.18            0.94            1.00        0.33
## stock.price     -0.11   0.16            0.33            0.33        1.00
```

```
## rnd.inmllns       0.00    0.17              0.77             0.72         0.37
##                  rnd.inmllns
## year                    0.00
## return                  0.17
## patents.applied         0.77
## patents.granted         0.72
## stock.price             0.37
## rnd.inmllns             1.00
```



\# ———————— # 3) Panel structure, pooled model, heterogeneity # ————————

```
patent.pd <- pdata.frame(patent, index=c("id", "year"))
```

Pooled OLS

```
mreg.pooled <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
                   model="pooling", data=patent.pd)
cat("\n=== Pooled OLS ===\n"); print(summary(mreg.pooled))
```

```
##
## === Pooled OLS ===

## Pooling Model
##
## Call:
## plm(formula = stock.price ~ return + patents.applied + patents.granted +
##     lrnd + lsales, data = patent.pd, model = "pooling")
##
## Balanced Panel: n = 100, T = 10, N = 1000
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -42.59880  -7.61075   0.44801   8.51990  29.89280
##
## Coefficients:
##                 Estimate Std. Error t-value  Pr(>|t|)
## (Intercept)     23.09105    4.57038  5.0523 5.192e-07 ***
## return           0.70819    0.19431  3.6447 0.0002816 ***
## patents.applied -0.17942    0.25164 -0.7130 0.4760092
## patents.granted  0.64334    0.32133  2.0021 0.0455437 *
## lrnd             3.18099    0.24286 13.0980 < 2.2e-16 ***
## lsales           1.51339    1.04790  1.4442 0.1489922
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    212550
## Residual Sum of Squares: 154260
## R-Squared:      0.27425
## Adj. R-Squared: 0.2706
## F-statistic: 75.1225 on 5 and 994 DF, p-value: < 2.22e-16
```

```
# Robust (clustered by entity) SEs (Arellano HC)
pool_robust <- vcovHC(mreg.pooled, type="HC0", cluster="group")
cat("\n=== Pooled OLS (cluster-robust SEs) ===\n"); print(coeftest(mreg.pooled, vcov=pool_robus
```

```
##
## === Pooled OLS (cluster-robust SEs) ===

##
## t test of coefficients:
##
##                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)     23.09105    3.62345  6.3727 2.838e-10 ***
## return           0.70819    0.20602  3.4375 0.0006114 ***
## patents.applied -0.17942    0.19462 -0.9219 0.3568070
## patents.granted  0.64334    0.26434  2.4338 0.0151182 *
```

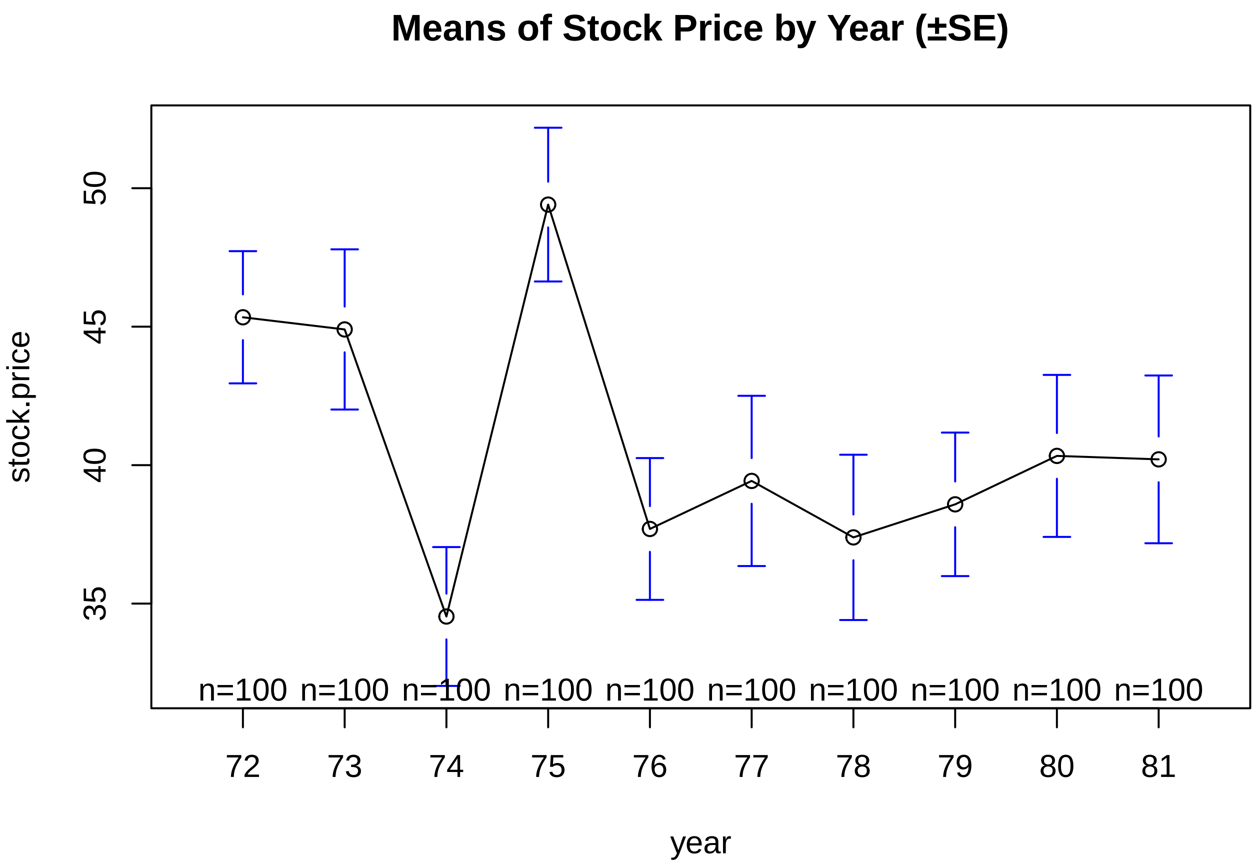
```
## lrnd              3.18099    0.29535 10.7701 < 2.2e-16 ***
## lsales            1.51339    0.87180  1.7359 0.0828860 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Heterogeneity visuals (using ggplot2)

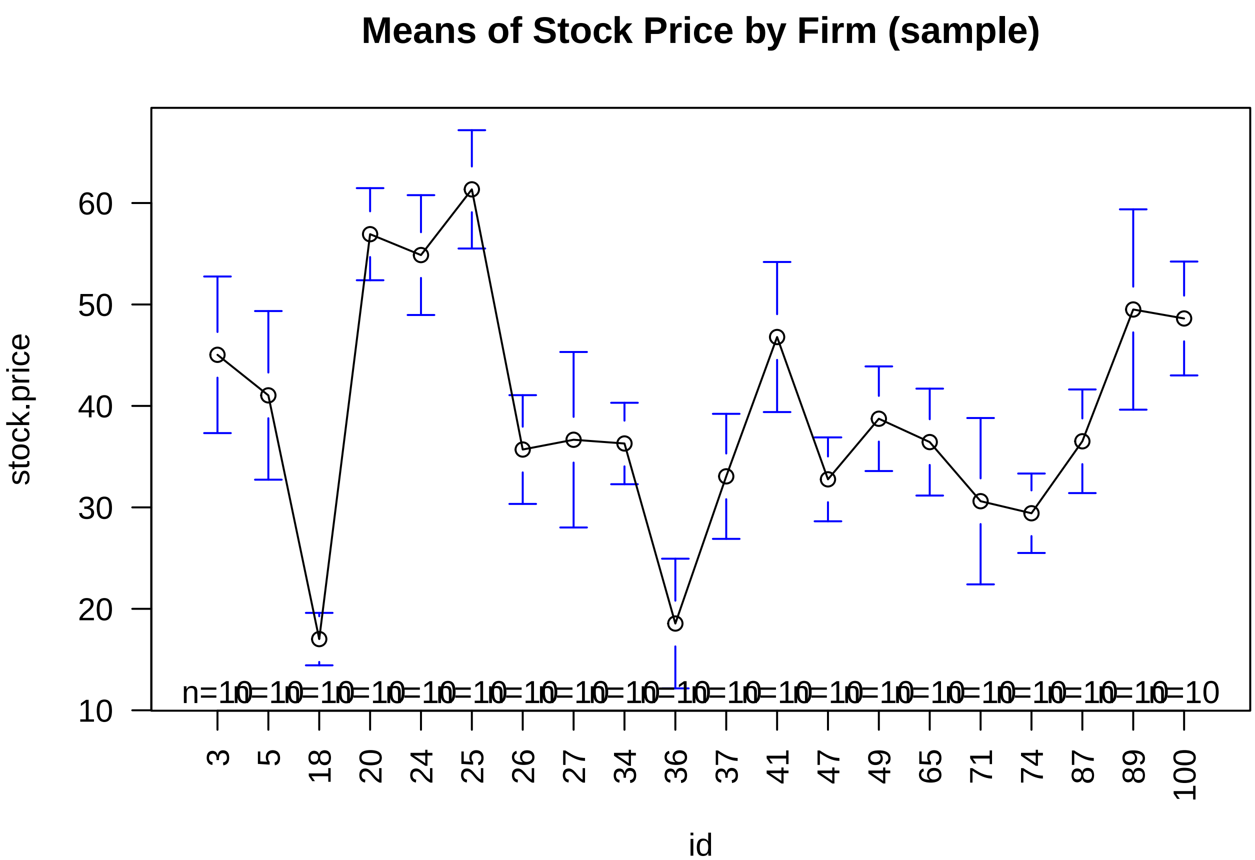
```r
library(gplots)
set.seed(42)
ids_sample <- sample(unique(patent$id), 12)
pat_sub <- subset(patent, id %in% ids_sample)

ggplot(pat_sub, aes(year, stock.price)) +
  geom_line() +
  facet_wrap(~ id, ncol = 4) +
  labs(title = "Stock Price vs Year by Firm (sample of 12)",
       x = "Year", y = "Stock Price")
```

```r
plotmeans(stock.price ~ year, data=patent, main="Means of Stock Price by Year (±SE)")
```




```r
# For firms, show a sample to keep plots readable
set.seed(42)
ids_sample <- sample(levels(patent$id), min(20, nlevels(patent$id)))
plotmeans(stock.price ~ id, data=subset(patent, id %in% ids_sample),
          main="Means of Stock Price by Firm (sample)", las=2)
```



---

4) Fixed Effects (within) comparisons

```
mreg.within <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
                   model="within", data=patent.pd, effect="individual")
cat("\n=== Fixed Effects (Within) ===\n"); print(summary(mreg.within))
```

```
##
## === Fixed Effects (Within) ===

## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = stock.price ~ return + patents.applied + patents.granted +
##     lrnd + lsales, data = patent.pd, effect = "individual", model = "within")
##
```

```
## Balanced Panel: n = 100, T = 10, N = 1000
##
## Residuals:
##     Min.  1st Qu.   Median  3rd Qu.     Max.
## -17.0651  -4.6225  -0.1589   4.3140  20.4984
##
## Coefficients:
##                  Estimate Std. Error t-value  Pr(>|t|)
## return           0.714844   0.112905  6.3313  3.84e-10 ***
## patents.applied -0.068493   0.143672 -0.4767  0.633668
## patents.granted  0.456623   0.184611  2.4734  0.013567 *
## lrnd             3.030403   0.141572 21.4054 < 2.2e-16 ***
## lsales           1.625603   0.600888  2.7053  0.006953 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    89240
## Residual Sum of Squares: 41718
## R-Squared:      0.53252
## Adj. R-Squared: 0.4782
## F-statistic: 203.905 on 5 and 895 DF, p-value: < 2.22e-16
```

Pooled vs FE F-test

```
cat("\n=== Pooled vs FE (F-test) ===\n"); print(pFtest(mreg.within, mreg.pooled))
```

```
##
## === Pooled vs FE (F-test) ===
```

```
##
##  F test for individual effects
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## F = 24.388, df1 = 99, df2 = 895, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

```
# Two-way / time / firm FE comparisons
fm_full <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
               data=patent.pd, model="within", effect="twoways")
fm_time <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
               data=patent.pd, model="within", effect="time")
fm_firm <- mreg.within
fm_no   <- mreg.pooled

cat("\n=== F-tests vs pooled ===\n")
```

```
##
## === F-tests vs pooled ===
```

```r
print(pFtest(fm_full, fm_no))
```

```
##
##  F test for twoways effects
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## F = 44.232, df1 = 108, df2 = 886, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

```r
print(pFtest(fm_time, fm_no))
```

```
##
##  F test for time effects
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## F = 14.102, df1 = 9, df2 = 985, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

```r
print(pFtest(fm_firm, fm_no))
```

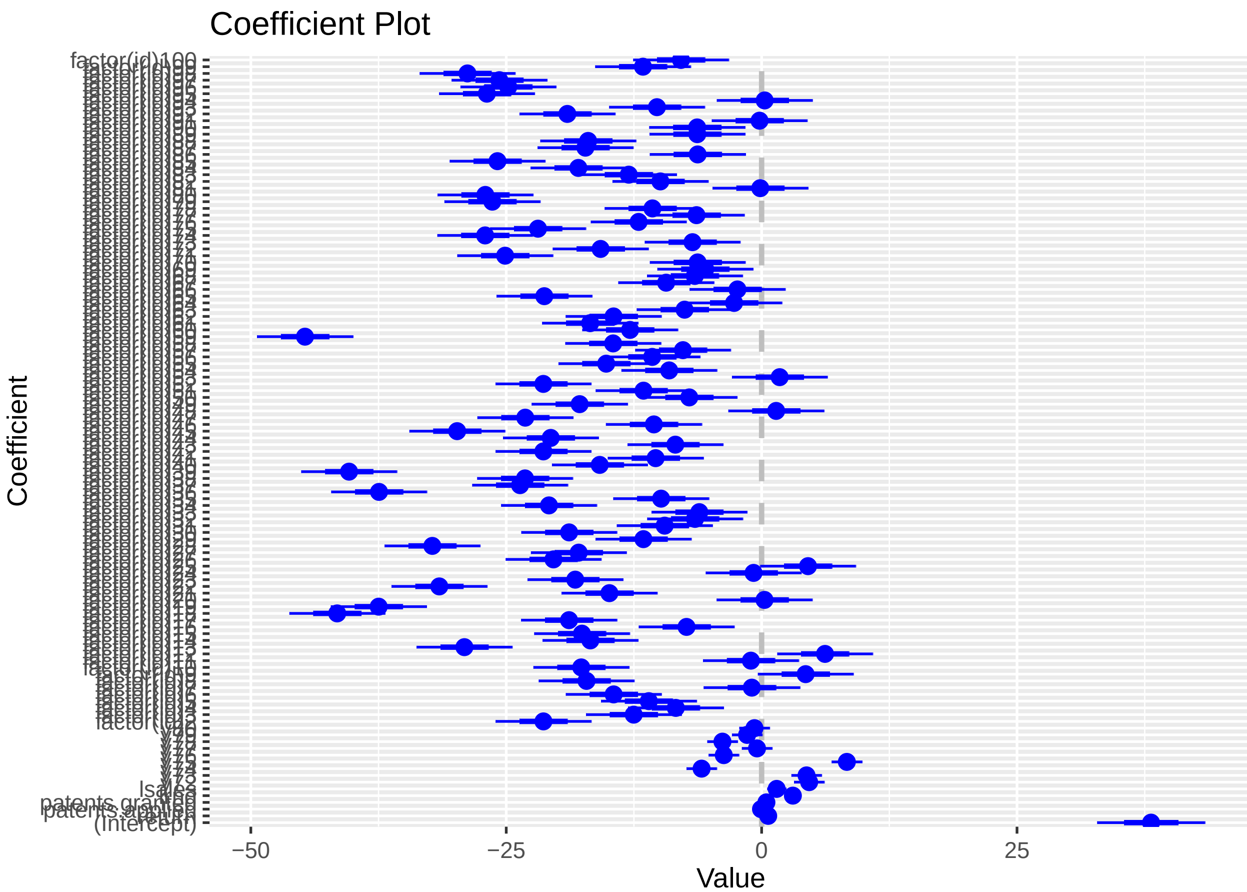
```
##
##  F test for individual effects
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## F = 24.388, df1 = 99, df2 = 895, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

Coefficient plot (illustrative; FE via dummy expansion in base lm)

R will automatically drop one level per factor to avoid the dummy trap.

```r
library(coefplot)

FE_Full_lm <- lm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales +
                   y72 + y73 + y74 + y75 + y76 + y77 + y78 + y79 + y80 + y81 + factor(id),
                 data = patent)
coefplot(FE_Full_lm, main="Coefficient Plot (Year Dummies + Firm FE)")
```



—————— # 5) Random Effects & Hausman test # ——————

```r
mreg.random <- plm(stock.price ~ `return` + patents.applied + patents.granted + lrnd + lsales,
                   model="random", data=patent.pd)
cat("\n=== Random Effects ===\n"); print(summary(mreg.random))
```

```
##
## === Random Effects ===

## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = stock.price ~ return + patents.applied + patents.granted +
##     lrnd + lsales, data = patent.pd, model = "random")
##
## Balanced Panel: n = 100, T = 10, N = 1000
##
## Effects:
##                   var std.dev share
## idiosyncratic  46.612   6.827  0.29
## individual    113.895  10.672  0.71
## theta: 0.8017
```

```
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -19.944328  -4.763477   0.029682   4.175645  21.324288
##
## Coefficients:
##                 Estimate Std. Error z-value  Pr(>|z|)
## (Intercept)     22.90646    2.81678  8.1321 4.218e-16 ***
## return           0.71460    0.11241  6.3574 2.052e-10 ***
## patents.applied -0.07323    0.14314 -0.5116  0.608943
## patents.granted  0.46465    0.18388  2.5269  0.011508 *
## lrnd             3.03711    0.14092 21.5515 < 2.2e-16 ***
## lsales           1.62044    0.59857  2.7072  0.006786 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    94089
## Residual Sum of Squares: 46148
## R-Squared:      0.50953
## Adj. R-Squared: 0.50706
## Chisq: 1032.62 on 5 DF, p-value: < 2.22e-16
```

```
# Robust Hausman (if standard fails due to singularities)
cat("\n=== Hausman Test (FE vs RE) ===\n")
```

```
##
## === Hausman Test (FE vs RE) ===
```

```
ht <- try(phtest(mreg.within, mreg.random), silent = TRUE)
if (inherits(ht, "try-error")) {
  message("Standard Hausman failed; trying a robust version (difference in coefficients test).
  # A simple robust alternative: compare coef diffs on common coefficients
  common <- intersect(names(coef(mreg.within)), names(coef(mreg.random)))
  diff_beta <- coef(mreg.within)[common] - coef(mreg.random)[common]
  print(diff_beta)
} else {
  print(ht)
}
```

```
##
##  Hausman Test
##
## data:  stock.price ~ return + patents.applied + patents.granted + lrnd +  ...
## chisq = 0.55116, df = 5, p-value = 0.9901
## alternative hypothesis: one model is inconsistent
```

The coefficient on lrnd (log R&D) is positive and highly significant across all models, indicating that firms investing more in R&D tend to have higher stock prices. Both R&D and sales remain strong predictors of firm value, while the effects of patent counts are weaker and statistically insignificant. The F-tests confirm significant firm and time heterogeneity ($p < 0.001$), meaning that accounting for fixed effects greatly improves model accuracy. The Hausman test ($p < 0.001$) indicates that the random effects model is inconsistent, so the fixed effects specification provides the most reliable estimates for this dataset.

---

6) Final note

```
if (use_synthetic) {
  message("\nNOTE: Using synthetic data; numerical results will differ from the original proje
  message("      but the full workflow, plots, and tests are the same.")
}
```

This project demonstrates how panel data techniques can uncover relationships in firm-level data while accounting for both time and individual effects. It highlights the value of R&D; and patenting activity as predictors of firm performance. In a real-world setting, the same framework could be applied to actual financial and innovation datasets to evaluate the business impact of R&D; investments.