Panel Data Analysis in R

Ajinkya

2024-12-24

Introduction

The purpose of the analysis is to evaluate the impact of two types of treatments (A and B) on reducing diastolic blood pressure (DBP) using panel data regression models, We use including both fixed and random effects to compare the treatments over time.

#Import data to R

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.3.3
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
## Warning: package 'tidyr' was built under R version 4.3.3
```

```
## Warning: package 'dplyr' was built under R version 4.4.0
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.1     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to be
come errors
```

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.3.3
```

```
dia_bp_data <- read_csv("C:/Users/Ajinkyaa/OneDrive/Stata to R/New folder/My_First_Project/Di
a_BP.csv", col_names = T)
```

```
## Rows: 100 Columns: 14
## ── Column specification ────────────────────────────────────────────────────
## Delimiter: ","
## chr (7): treatment, sex, hypertension_1, hypertension_2, hypertension_3, hyp...
## dbl (7): id, dbp_1, dbp_2, dbp_3, dbp_4, dbp_5, age
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

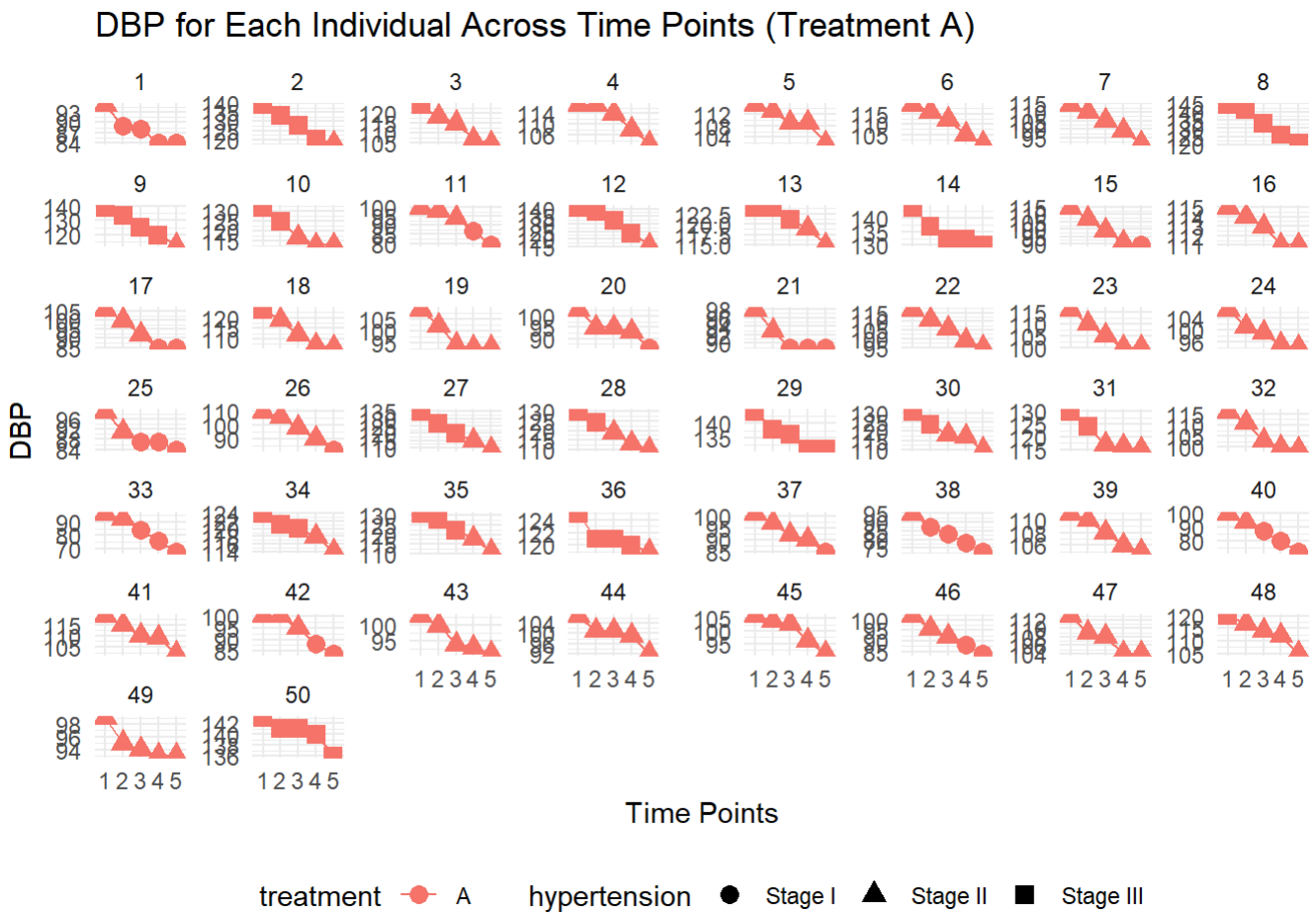
```
head(dia_bp_data)
```

```
## # A tibble: 6 × 14
##      id treatment dbp_1 dbp_2 dbp_3 dbp_4 dbp_5   age sex   hypertension_1
##   <dbl> <chr>     <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <chr> <chr>
## 1     1 A            95    89    88    84    84    48 F     Stage II
## 2     2 A           139   133   128   120   118    65 F     Stage III
## 3     3 A           124   118   115   107   105    55 M     Stage III
## 4     4 A           117   117   114   108   103    56 M     Stage II
## 5     5 A           115   113   109   109   102    44 F     Stage II
## 6     6 A           119   116   112   105   101    67 M     Stage II
## # ℹ 4 more variables: hypertension_2 <chr>, hypertension_3 <chr>,
## #   hypertension_4 <chr>, hypertension_5 <chr>
```

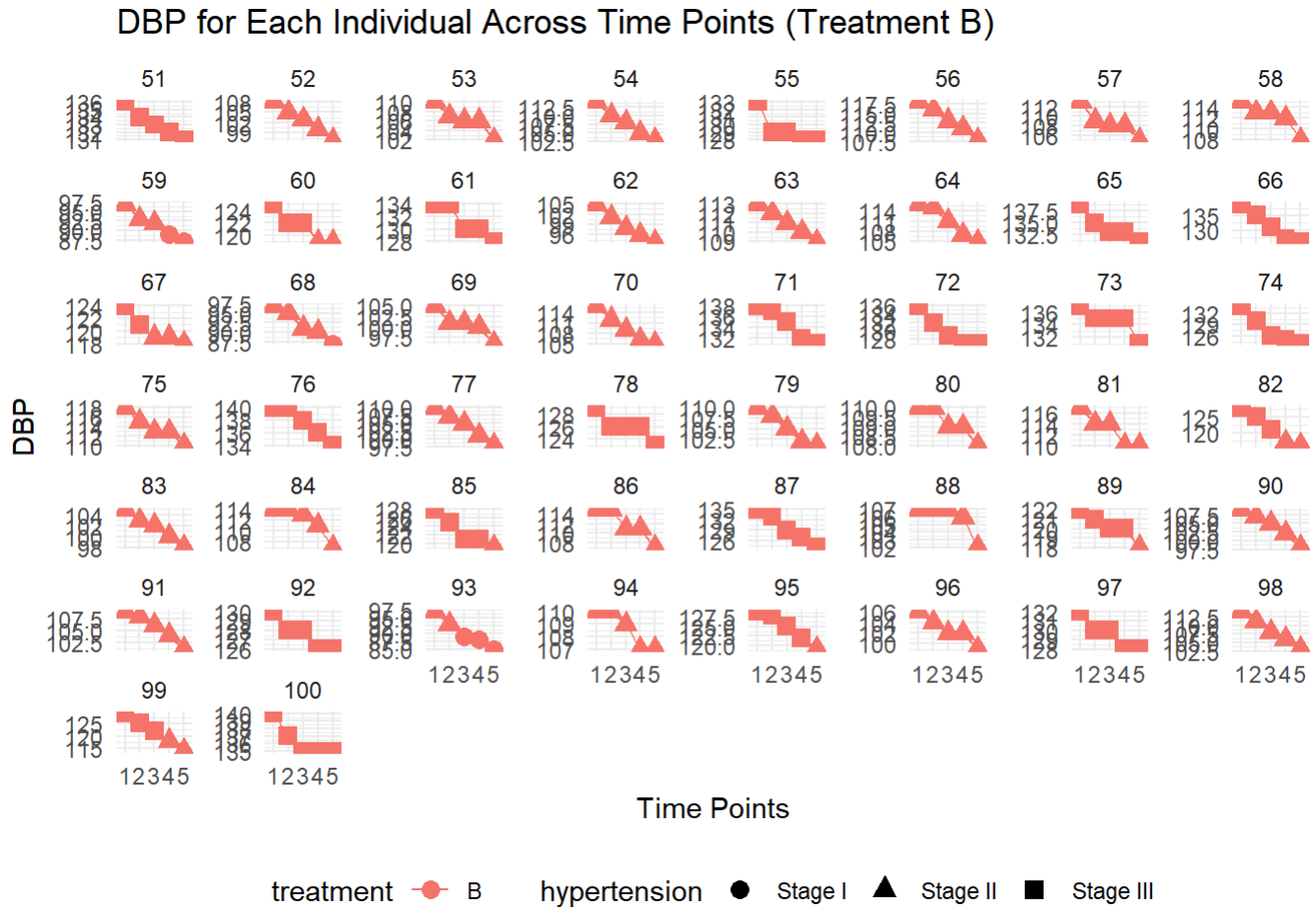
```
# Reshape the data
reshaped_data <- dia_bp_data %>%
  pivot_longer(cols = starts_with("dbp"),
               names_to = "time",
               values_to = "dbp") %>%
  pivot_longer(cols = starts_with("hypertension"),
               names_to = "time_hypertension",
               values_to = "hypertension") %>%
  separate(time, into = c("type", "num"), sep = "_") %>%
  separate(time_hypertension, into = c("type_hypertension", "num_hypertension"), sep = "_") %
>%
  filter(num == num_hypertension)

# View the reshaped data
print(reshaped_data)
```

```
## # A tibble: 500 × 10
##       id treatment   age sex   type  num     dbp type_hypertension
##    <dbl> <chr>     <dbl> <chr> <chr> <chr> <dbl> <chr>
##  1     1 A            48 F     dbp   1        95 hypertension
##  2     1 A            48 F     dbp   2        89 hypertension
##  3     1 A            48 F     dbp   3        88 hypertension
##  4     1 A            48 F     dbp   4        84 hypertension
##  5     1 A            48 F     dbp   5        84 hypertension
##  6     2 A            65 F     dbp   1       139 hypertension
##  7     2 A            65 F     dbp   2       133 hypertension
##  8     2 A            65 F     dbp   3       128 hypertension
##  9     2 A            65 F     dbp   4       120 hypertension
## 10     2 A            65 F     dbp   5       118 hypertension
## # ℹ 490 more rows
## # ℹ 2 more variables: num_hypertension <chr>, hypertension <chr>
```

```
#Plot the Diastolic BP and type of hypertension of each individual for the all the time point
s.

library(ggplot2)

# Filter the data for treatment A
treatment_A_data <- reshaped_data %>% filter(treatment == "A")

# Plot for treatment A
plot_A <- ggplot(treatment_A_data, aes(x = num, y = dbp, group = id, color = treatment)) +
  geom_line() +  # Line plot to show DBP trend over time
  geom_point(aes(shape = hypertension), size = 3) +  # Add points with different shapes for h
ypertension stages
  labs(x = "Time Points", y = "DBP", title = "DBP for Each Individual Across Time Points (Tre
atment A)") +
  facet_wrap(~ id, scales = "free_y") +  # Facet by individual (id)
  theme_minimal() +  # Minimal theme for better appearance
  theme(legend.position = "bottom")  # Position legend at the bottom

# Print plot for treatment A
print(plot_A)
```



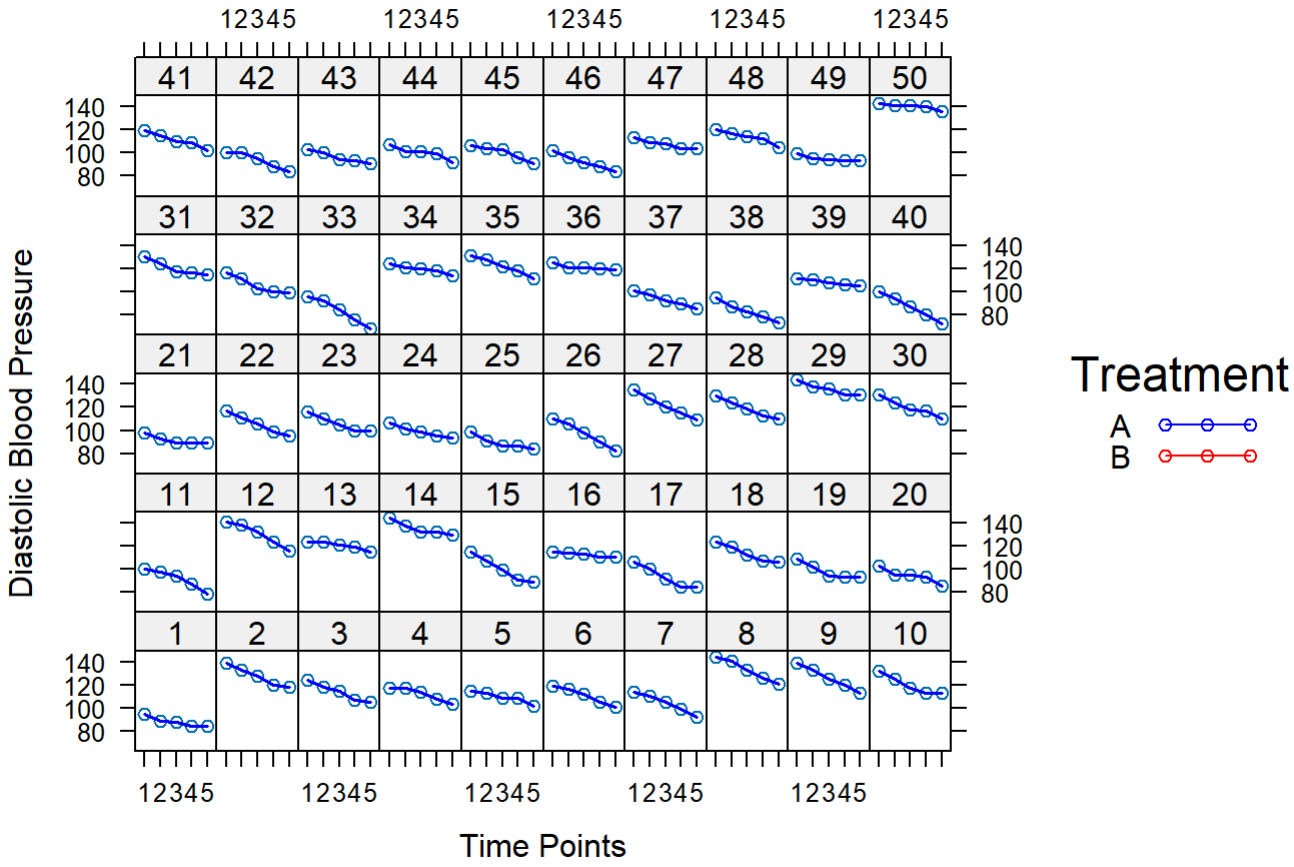

```
# Filter the data for treatment B
treatment_B_data <- reshaped_data %>% filter(treatment == "B")

# Plot for treatment B
plot_B <- ggplot(treatment_B_data, aes(x = num, y = dbp, group = id, color = treatment)) +
  geom_line() +  # Line plot to show DBP trend over time
  geom_point(aes(shape = hypertension), size = 3) +  # Add points with different shapes for h
ypertension stages
  labs(x = "Time Points", y = "DBP", title = "DBP for Each Individual Across Time Points (Tre
atment B)") +
  facet_wrap(~ id, scales = "free_y") +  # Facet by individual (id)
  theme_minimal() +  # Minimal theme for better appearance
  theme(legend.position = "bottom")  # Position legend at the bottom

# Print plot for treatment B
print(plot_B)
```

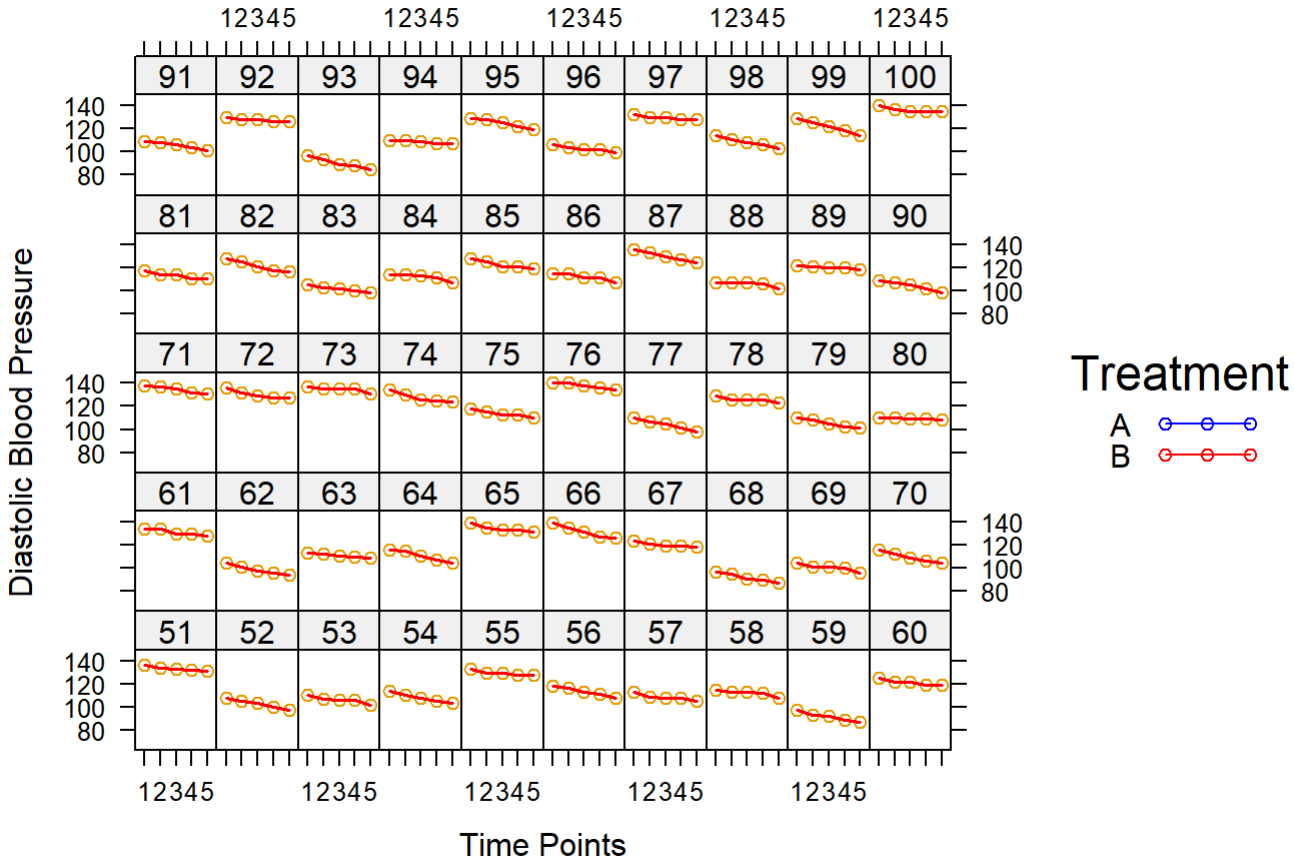



```
# install.packages("lattice")
library(lattice)
```

```
## Warning: package 'lattice' was built under R version 4.3.3
```

```
xyplot(dbp ~ as.factor(num) | factor(id),
       data = reshaped_data,
       type = "o",
       lwd = 1.5,
       layout = c(10, 5),
       xlab = "Time Points",
       ylab = "Diastolic Blood Pressure",
       groups = reshaped_data$treatment, # Color lines by treatment group'
       auto.key = list(space = "right", title = "Treatment", lines = TRUE, points = FALSE),
       par.settings = list(superpose.line = list(col = c("blue", "red"))) # Define colors for
the groups
)
```


Diastolic Blood Pressure
Time Points
Treatment
A
B
41 42 43 44 45 46 47 48 49 50
31 32 33 34 35 36 37 38 39 40
21 22 23 24 25 26 27 28 29 30
11 12 13 14 15 16 17 18 19 20
1 2 3 4 5 6 7 8 9 10
12345
140
120
100
80

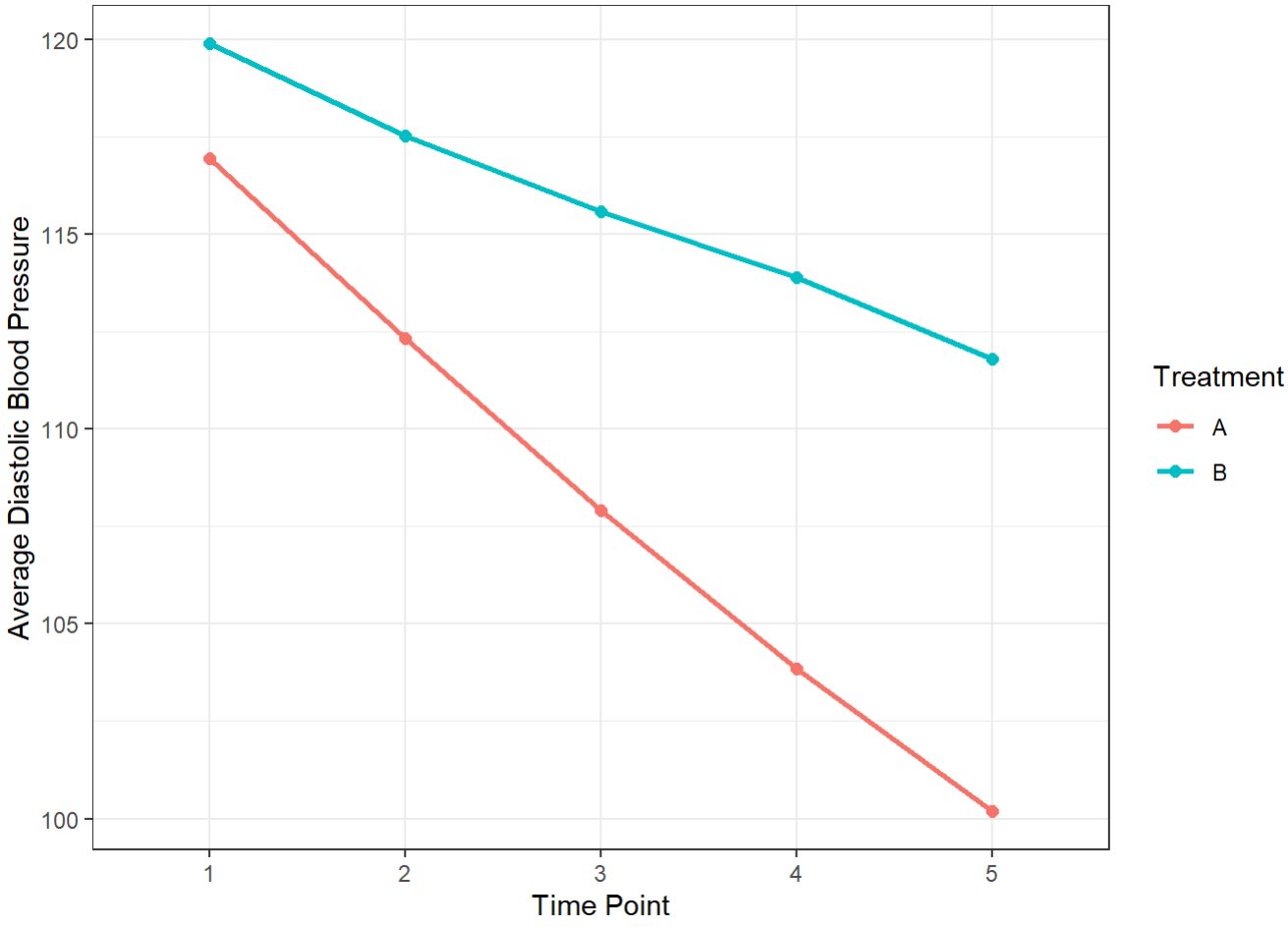

Diastolic Blood Pressure
Time Points
Treatment
A
B
91 92 93 94 95 96 97 98 99 100
81 82 83 84 85 86 87 88 89 90
71 72 73 74 75 76 77 78 79 80
61 62 63 64 65 66 67 68 69 70
51 52 53 54 55 56 57 58 59 60
12345
140
120
100
80

```
# Plot the mean DBP at the 5 time points separately for treatment A and B and plot it against time.
mean_dbp <- reshaped_data %>%
  group_by(num, treatment) %>%
  summarise(meandbp = mean(dbp)) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'num'. You can override using the `.groups`
## argument.
```

```
ggplot(mean_dbp, aes(x = num, y = meandbp, color = treatment, group = treatment)) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  labs(x = "Time Point",
       y = "Average Diastolic Blood Pressure",
       color = "Treatment") +                 # Label the plot
  theme_bw()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```
# Load the necessary package
#install.packages("plm")
library(plm)
```

```
## Warning: package 'plm' was built under R version 4.3.3
```

```
##
## Attaching package: 'plm'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, lag, lead
```

```
# Fit the fixed effects model
model1.fe <- plm(dbp ~ num, data = reshaped_data, index = c("id", "num"),
             model = "within")

# Summarize the model
summary(model1.fe)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = dbp ~ num, data = reshaped_data, model = "within",
##     index = c("id", "num"))
##
## Balanced Panel: n = 100, T = 5, N = 500
##
## Residuals:
##    Min. 1st Qu.  Median 3rd Qu.    Max.
##  -9.194  -1.624   0.176   1.516   8.376
##
## Coefficients:
##       Estimate Std. Error  t-value  Pr(>|t|)
## num2  -3.48000    0.41341  -8.4179 7.094e-16 ***
## num3  -6.67000    0.41341 -16.1343 < 2.2e-16 ***
## num4  -9.54000    0.41341 -23.0766 < 2.2e-16 ***
## num5 -12.43000    0.41341 -30.0673 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    12962
## Residual Sum of Squares: 3383.9
## R-Squared:      0.73893
## Adj. R-Squared: 0.67102
## F-statistic: 280.206 on 4 and 396 DF, p-value: < 2.22e-16
```

```
# Interaction between time and treatment
# Treatment "B" as the reference

reshaped_data$treatment <- relevel(factor(reshaped_data$treatment), ref = "B")

#running the model with interaction term
model2.fe <- plm(dbp ~ num*treatment, data = reshaped_data, index = c("id", "num"),
                 model = "within")
summary(model2.fe)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = dbp ~ num * treatment, data = reshaped_data, model = "within",
##     index = c("id", "num"))
##
## Balanced Panel: n = 100, T = 5, N = 500
##
## Residuals:
##    Min. 1st Qu.  Median 3rd Qu.    Max.
##  -7.132  -1.156   0.068   1.244   6.268
##
## Coefficients:
##                 Estimate Std. Error  t-value  Pr(>|t|)
## num2            -2.36000    0.46840  -5.0385 7.178e-07 ***
## num3            -4.32000    0.46840  -9.2229 < 2.2e-16 ***
## num4            -6.00000    0.46840 -12.8096 < 2.2e-16 ***
## num5            -8.10000    0.46840 -17.2930 < 2.2e-16 ***
## num2:treatmentA -2.24000    0.66241  -3.3816 0.0007932 ***
## num3:treatmentA -4.70000    0.66241  -7.0953 6.082e-12 ***
## num4:treatmentA -7.08000    0.66241 -10.6882 < 2.2e-16 ***
## num5:treatmentA -8.66000    0.66241 -13.0734 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    12962
## Residual Sum of Squares: 2150.1
## R-Squared:      0.83412
## Adj. R-Squared: 0.78884
## F-statistic: 246.393 on 8 and 392 DF, p-value: < 2.22e-16
```

```
# Random Effect Regression Model
# with time and treatment as co variate

model1.re <- plm(dbp ~ num + treatment, data = reshaped_data, index = c("id", "num"),
             model = "random")
summary(model1.re)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = dbp ~ num + treatment, data = reshaped_data, model = "random",
##     index = c("id", "num"))
##
## Balanced Panel: n = 100, T = 5, N = 500
##
## Effects:
##                   var std.dev share
## idiosyncratic   8.545   2.923 0.043
## individual    191.153  13.826 0.957
## theta: 0.9059
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -11.55190  -1.81926  -0.12757   1.83600   8.59524
##
## Coefficients:
##              Estimate Std. Error  z-value  Pr(>|z|)
## (Intercept) 122.16800    1.98132  61.6600 < 2.2e-16 ***
## num2         -3.48000    0.41341  -8.4179 < 2.2e-16 ***
## num3         -6.67000    0.41341 -16.1343 < 2.2e-16 ***
## num4         -9.54000    0.41341 -23.0766 < 2.2e-16 ***
## num5        -12.43000    0.41341 -30.0673 < 2.2e-16 ***
## treatmentA   -7.49600    2.77750  -2.6988  0.006958 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    13861
## Residual Sum of Squares: 4221.3
## R-Squared:      0.69546
## Adj. R-Squared: 0.69238
## Chisq: 1128.11 on 5 DF, p-value: < 2.22e-16
```

```
# Interaction between time and treatment
model2.re <- plm(dbp ~ num*treatment, data = reshaped_data, index = c("id", "num"),
             model = "random")
summary(model2.re)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = dbp ~ num * treatment, data = reshaped_data, model = "random",
##     index = c("id", "num"))
##
## Balanced Panel: n = 100, T = 5, N = 500
##
## Effects:
##                   var std.dev share
## idiosyncratic   5.485   2.342 0.028
## individual    191.766  13.848 0.972
## theta: 0.9246
##
## Residuals:
##     Min.  1st Qu.   Median  3rd Qu.     Max.
## -9.02107 -1.39573 -0.10514  1.47568  6.84481
##
## Coefficients:
##                  Estimate Std. Error  z-value  Pr(>|z|)
## (Intercept)     119.90000    1.98620  60.3664 < 2.2e-16 ***
## num2             -2.36000    0.46840  -5.0385 4.693e-07 ***
## num3             -4.32000    0.46840  -9.2229 < 2.2e-16 ***
## num4             -6.00000    0.46840 -12.8096 < 2.2e-16 ***
## num5             -8.10000    0.46840 -17.2930 < 2.2e-16 ***
## treatmentA       -2.96000    2.80892  -1.0538 0.2919806
## num2:treatmentA  -2.24000    0.66241  -3.3816 0.0007207 ***
## num3:treatmentA  -4.70000    0.66241  -7.0953 1.291e-12 ***
## num4:treatmentA  -7.08000    0.66241 -10.6882 < 2.2e-16 ***
## num5:treatmentA  -8.66000    0.66241 -13.0734 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    13539
## Residual Sum of Squares: 2687.6
## R-Squared:      0.80149
## Adj. R-Squared: 0.79785
## Chisq: 1978.43 on 9 DF, p-value: < 2.22e-16
```

```
# sex and age as covariates
model3.re <- plm(dbp ~ num*treatment + age + sex, data = reshaped_data, index = c("id", "nu
m"),
             model = "random")
summary(model3.re)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = dbp ~ num * treatment + age + sex, data = reshaped_data,
##     model = "random", index = c("id", "num"))
##
## Balanced Panel: n = 100, T = 5, N = 500
##
## Effects:
##                   var std.dev share
## idiosyncratic   5.485   2.342  0.03
## individual    178.547  13.362  0.97
## theta: 0.9219
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -8.806714 -1.366682 -0.049822  1.499243  6.537325
##
## Coefficients:
##                   Estimate Std. Error  z-value  Pr(>|z|)
## (Intercept)     117.728656   8.131410  14.4783 < 2.2e-16 ***
## num2             -2.360000   0.468397  -5.0385 4.693e-07 ***
## num3             -4.320000   0.468397  -9.2229 < 2.2e-16 ***
## num4             -6.000000   0.468397 -12.8096 < 2.2e-16 ***
## num5             -8.100000   0.468397 -17.2930 < 2.2e-16 ***
## treatmentA       -3.094432   2.717329  -1.1388 0.2547962
## age              -0.029132   0.144794  -0.2012 0.8405433
## sexM              8.090804   2.694562   3.0026 0.0026765 **
## num2:treatmentA  -2.240000   0.662414  -3.3816 0.0007207 ***
## num3:treatmentA  -4.700000   0.662414  -7.0953 1.291e-12 ***
## num4:treatmentA  -7.080000   0.662414 -10.6882 < 2.2e-16 ***
## num5:treatmentA  -8.660000   0.662414 -13.0734 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    13582
## Residual Sum of Squares: 2676.6
## R-Squared:      0.80292
## Adj. R-Squared: 0.79848
## Chisq: 1988.17 on 11 DF, p-value: < 2.22e-16
```

```
# Hausman Test Example
data("Grunfeld", package = "plm")


# Fixed effects model
fe_model <- plm(inv ~ value + capital, data = Grunfeld, model = "within")

# Random effects model
re_model <- plm(inv ~ value + capital, data = Grunfeld, model = "random")

# Hausman test
hausman_test <- phtest(model2.fe, model3.re)
print(hausman_test)
```

```
##
##  Hausman Test
##
## data:  dbp ~ num * treatment
## chisq = 9.7718e-12, df = 8, p-value = 1
## alternative hypothesis: one model is inconsistent
```

Null Hypothesis: The random effects model is consistent and efficient.

Alternative Hypothesis: The random effects model is inconsistent.

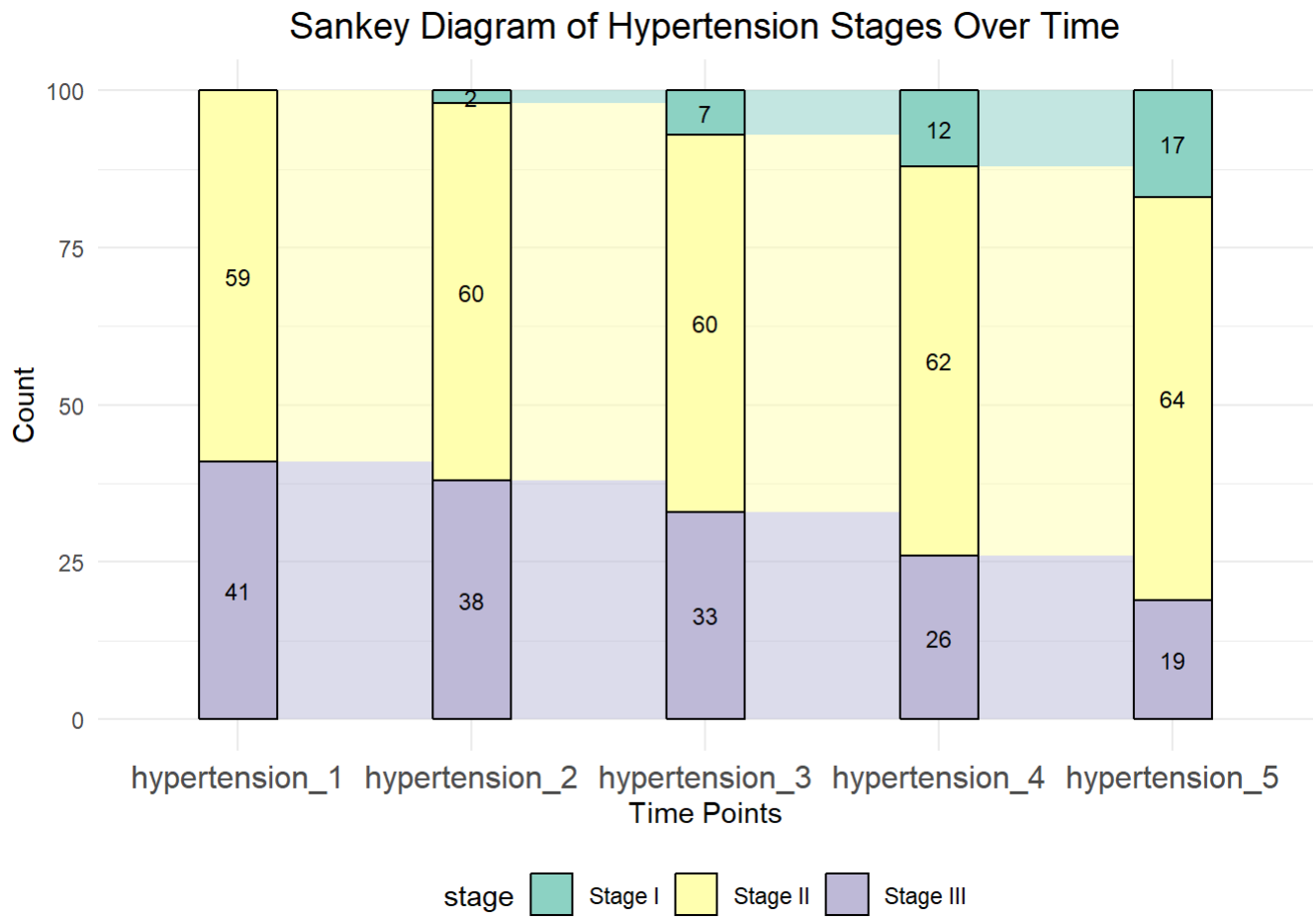
p-value: The p-value is extremely small (< 2.2e-16), far smaller than any conventional significance level (such as 0.05 or 0.01). This indicates strong evidence against the null hypothesis.

Interpretation: Since the p-value is very small, we reject the null hypothesis. This suggests that the random effects model is inconsistent, and the fixed effects model is preferred for this analysis. The reason for this conclusion is that there is significant evidence that the assumptions of the random effects model are violated, making the fixed effects model a more reliable choice for this dataset.

```
# Load necessary libraries
library(dplyr)
library(ggplot2)
library(ggalluvial)
```

```
## Warning: package 'ggalluvial' was built under R version 4.3.3
```

```
# Prepare the data for the Sankey diagram
sankey_data <- dia_bp_data %>%
  select(hypertension_1, hypertension_2, hypertension_3, hypertension_4, hypertension_5) %>%
  mutate(id = row_number()) %>%
  pivot_longer(cols = starts_with("hypertension"),
               names_to = "time_point", values_to = "stage") %>%
  mutate(time_point = factor(time_point,
                             levels = c("hypertension_1", "hypertension_2",
                                        "hypertension_3", "hypertension_4",
                                        "hypertension_5")))

# Add counts for clear flow
transition_counts <- sankey_data %>%
  group_by(time_point, stage) %>%
  summarise(count = n(), .groups = "drop")

# Plot Sankey diagram with numbers
ggplot(sankey_data, aes(x = time_point, stratum = stage, alluvium = id, fill = stage, label =
stage)) +
  geom_flow(stat = "alluvium", aes.bind = "alluvia") +
  geom_stratum() +
  geom_text(stat = "stratum", aes(label = after_stat(count)), size = 3, color = "black") +
  scale_fill_brewer(type = "qual", palette = "Set3") +
  theme_minimal() +
  labs(title = "Sankey Diagram of Hypertension Stages Over Time",
       x = "Time Points",
       y = "Count") +
  theme(legend.position = "bottom",
        plot.title = element_text(hjust = 0.5, size = 14),
        axis.text.x = element_text(size = 12))
```


Sankey Diagram of Hypertension Stages Over Time
Count
100
75
50
25
0
59
41
2
60
38
7
60
33
12
62
26
17
64
19
hypertension_1
hypertension_2
hypertension_3
hypertension_4
hypertension_5
Time Points
stage
Stage I
Stage II
Stage III