Module 5 Exercise - Part B

Your Name Here

The following is an exercise where the first two (data preparation) steps are already done for you. Complete steps 3 and 4 to visualize the data and perform regression analysis.

1. Create an extract of the nls97 dataset.

- Initialize the packages used in this analysis.

```
library(rio)
library(tidyverse)
library(plm)
library(ggplot2)
library(ggthemes)
library(broom)
library(lmtest)
library(stargazer)
```

- Load the final nlsy97 dataset extract (taken from the end of Module 2).

```
nlsy97 <- import("nlsy97.rds")
```

- Create the following new variables:
  - *logparentincome*, equal to the log of parent income.
  - A variable for the highest degree completed by the mother, equal to:
    - College if *motheredyrs* is greater than or equal to 14.
    - High school if *motheredyrs* is between 12 and 13.
    - Less than high school if *motheredyrs* is less than 12.

```
nlsy97 <- nlsy97 %>% mutate(logparentincome = log(parentincome))

nlsy97 <- nlsy97 %>% mutate(mother_degree = case_when(
  motheredyrs >= 14 ~ "college",
  motheredyrs %in% 12:13 ~ "high school",
  motheredyrs %in% 0:11 ~ "less than high school"
))
```

- Turn the mother's degree variable into an ordered factor.

```
nlsy97 <- nlsy97 %>%
  mutate(mother_degree = factor(mother_degree,
         levels = c("less than high school",
                    "high school",
                    "college"),
         ordered=TRUE))
```

2. Create a new dataset, nlsy97_sample, modified from nlsy97, which:

- Drops missing values in the following variables:
  - *parentincome*
  - *motheredyrs*

  - *gpa*
  - *highestgrade*

```
nlsy97_sample <- nlsy97 %>% drop_na(parentincome,motheredyrs,gpa,highestgrade)
# Another approach would be:
# nlsy97 <- nlsy97 %>% filter(!is.na(parentincome),
#                              !is.na(motheredyrs),
#                              !is.na(gpa),
#                              !is.na(highestgrade))
```

- Subsets the dataset for only observations where the student is 18 years old, with a GPA greater than or equal to 1.3 (D+ average), with parent income between $5,000 and $100,000.

```
nlsy97_sample <- nlsy97_sample %>% filter(age == 18,
                             gpa > 1.3,
                             parentincome > 5000,
                             parentincome < 100000)
```

3. Create a scatterplot

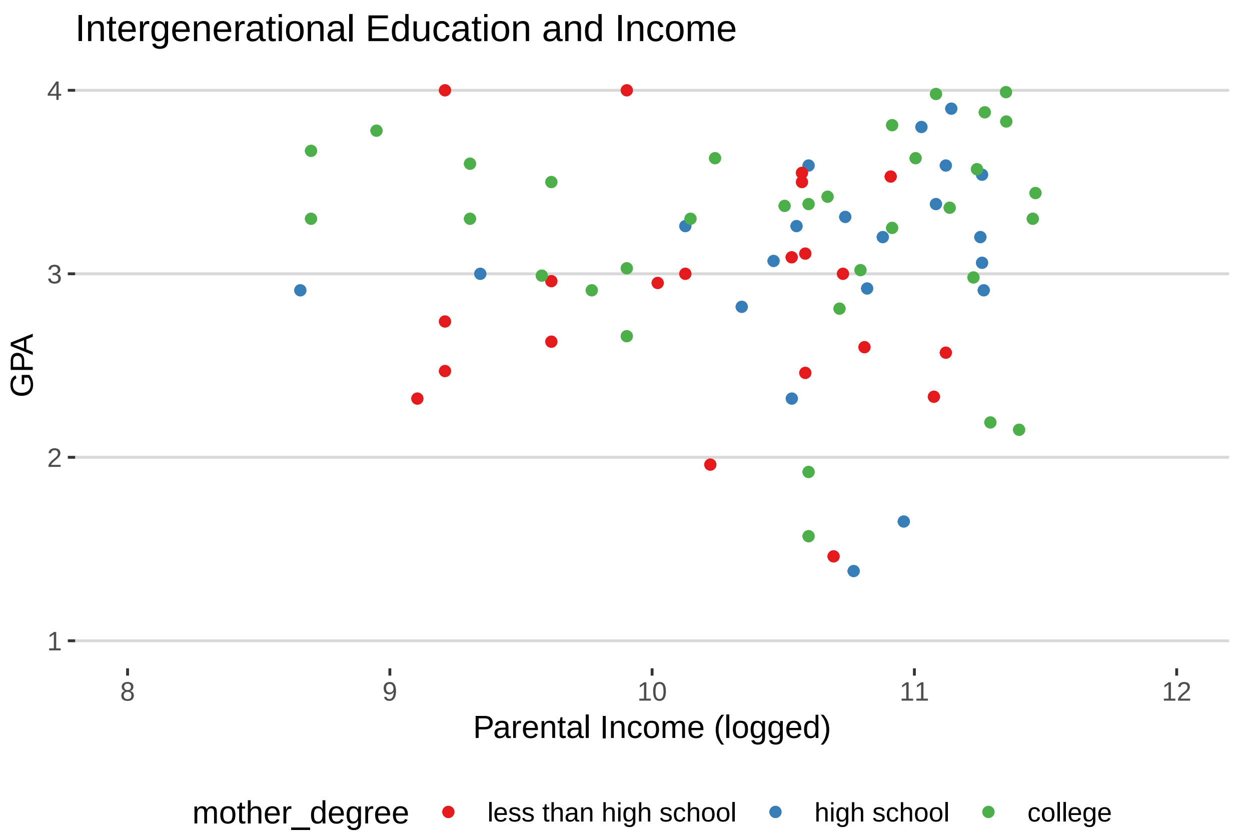
Create a scatterplot with:

- The dataset is 75 randomly sampled observations from **nlsy_sample**
- Log Parental income is on the x-axis, shown from values of 8 to 12.
- GPA is on the y-axis, shown from values 1 to 4.
- The color of points based on the mother's years of education.
- The color scale set using the following:

```
scale_colour_brewer(palette = "Set1")
```

- Add a title and axis labels.

```
ggplot(sample_n(nlsy97_sample,75),
       aes(x = logparentincome, y = gpa, col= mother_degree)) +
  geom_point() + scale_colour_brewer(palette = "Set1") +
  ggtitle("Intergenerational Education and Income") +
  xlab("Parental Income (logged)") + ylab("GPA") +
  ylim(1,4) + xlim(8,12) + theme_hc()
```



4. Perform a regression analysis of the effect of parent income on GPA

- Remove observations with Parental Income less than $5,000 and GPA less than or equal to 1.

```
nlsy97 <- nlsy97 %>% filter(parentincome>5000, gpa > 1)
```

- First run an OLS regression, with:
  - The log of GPA as the dependent variable
  - School type and log of parent income as the independent vaariables

```
gpa_and_parentincome <- lm(log(gpa) ~ log(parentincome) + schooltype, data = nlsy97)
```

- Display the results with tidy

```
tidy(gpa_and_parentincome)
```

term	estimate	std.error	statistic	p.value
(Intercept)	0.8094845	0.1140614	7.0969161	0.0000000
log(parentincome)	0.0217481	0.0106310	2.0457206	0.0411388
schooltype2	0.2632908	0.1460024	1.8033320	0.0717449
schooltype3	0.1012867	0.0516477	1.9611063	0.0502433
schooltype4	-0.0252134	0.0367681	-0.6857407	0.4930926

- Test for autocorrelation and report the statistical decision.

```
bgtest(gpa_and_parentincome)
```

```
##
```

```
##  Breusch-Godfrey test for serial correlation of order up to 1
##
## data:  gpa_and_parentincome
## LM test = 9.9264, df = 1, p-value = 0.001629
```

- Re-run the regression as fixed effects regression, including both unit and time fixed effects.

```
gpa_and_parentincome_fe <- plm(log(gpa) ~ log(parentincome) + schooltype, data = nlsy97,
                        index = c("personid","year"),
                        model = "within", effect="twoway")
```

- Perform autocorrelation-robust inference using the fixed effects regression
  - Test the coefficients of the model using the Stata-style HC_1 estimation of Newey-West heteroskedasticity and autocorrelated (HAC) robust standard errors.
  - Use the function vcovNW() for specifying the variance method inside of coeftest().
  - View the results with **tidy()**

```
tidy(coeftest(gpa_and_parentincome_fe,vcov =
           vcovNW(gpa_and_parentincome_fe,type="HC1")))
```

term	estimate	std.error	statistic	p.value
log(parentincome)	0.0957176	0.0461566	2.0737568	0.0427127
schooltype4	-0.0367467	0.1333158	-0.2756369	0.7838415