

# Class Survey Project

All Members of Table 12

2025-10-09

## Executive Summary

## Data Description

## Question of Interest

## Findings

#Basic Regression of the Class Survey Data # load data

```
csv_path <- "data/PSTAT197A_Standardized_Proficiency_Score.csv"
df <- readr::read_csv(csv_path, show_col_types = FALSE)

names(df)[names(df) == "Score_Avg (Difficulty of Courses)"] <- "Score_Avg"
```

make sure columns are numeric

```
df <- df %>%
  mutate(
    Score_Avg = as.numeric(Score_Avg),
    Prog_Proficiency = as.numeric(Prog_Proficiency),
    Stat_Proficiency = as.numeric(Stat_Proficiency),
    Math_Proficiency = as.numeric(Math_Proficiency)
  )
```

define pairs and fit models

```
pairs <- tribble(
  ~y, ~x,
  "Prog_Proficiency", "Score_Avg",
  "Stat_Proficiency", "Score_Avg",
  "Math_Proficiency", "Score_Avg"
```

```

)

fit_1 <- function(y, x, data) {
  sub <- data %>% select(all_of(c(y, x))) %>% drop_na()
  m <- lm(reformulate(x, y), data = sub)
  list(
    glance = glance(m) %>% mutate(y = y, x = x, n = nobs(m)),
    tidy = tidy(m) %>% mutate(y = y, x = x),
    model = m,
    data = sub
  )
}

fits <- pairs %>%
  pmap(fit_1, data = df)

```

## summarize coefficients & fit stats

```

coef_tbl <- map_dfr(fits, "tidy") %>%
  select(y, x, term, estimate, std.error, statistic, p.value)

glance_tbl <- map_dfr(fits, "glance") %>%
  select(y, x, n, r.squared, adj.r.squared, AIC, BIC, sigma, p.value)

write_csv(coef_tbl, "results/regression_coefficients.csv")
write_csv(glance_tbl, "results/regression_fit_stats.csv")

```

## Plots with fitted line

```

plot_one <- function(y, x, data) {
  rngx <- range(data[[x]], na.rm = TRUE) + c(-0.05, 0.05)
  rngy <- range(data[[y]], na.rm = TRUE) + c(-0.5, 0.5)

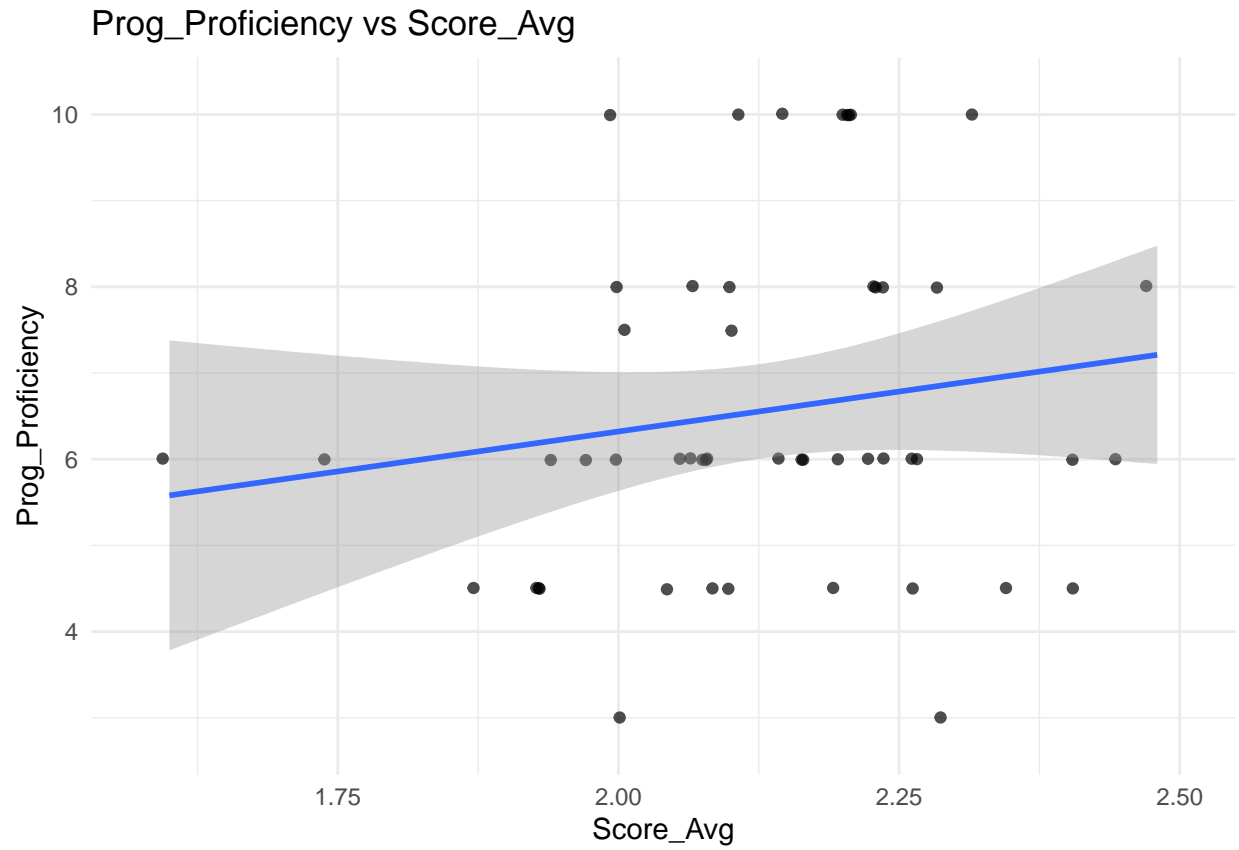
  ggplot(data, aes(x = .data[[x]], y = .data[[y]])) +
    geom_point(alpha = 0.7, position = position_jitter(width = 0.01, height = 0.01)) +
    geom_smooth(method = "lm", se = TRUE) +
    coord_cartesian(xlim = rngx, ylim = rngy) +
    scale_x_continuous(expand = expansion(mult = c(0.02, 0.02))) +
    scale_y_continuous(expand = expansion(mult = c(0.02, 0.02))) +
    labs(title = paste(y, "vs", x), x = x, y = y) +
    theme_minimal()
}

plots <- pmap(
  pairs,
  \(y, x) plot_one(y, x, df %>% select(all_of(c(y, x))) %>% tidyr::drop_na())
)

```

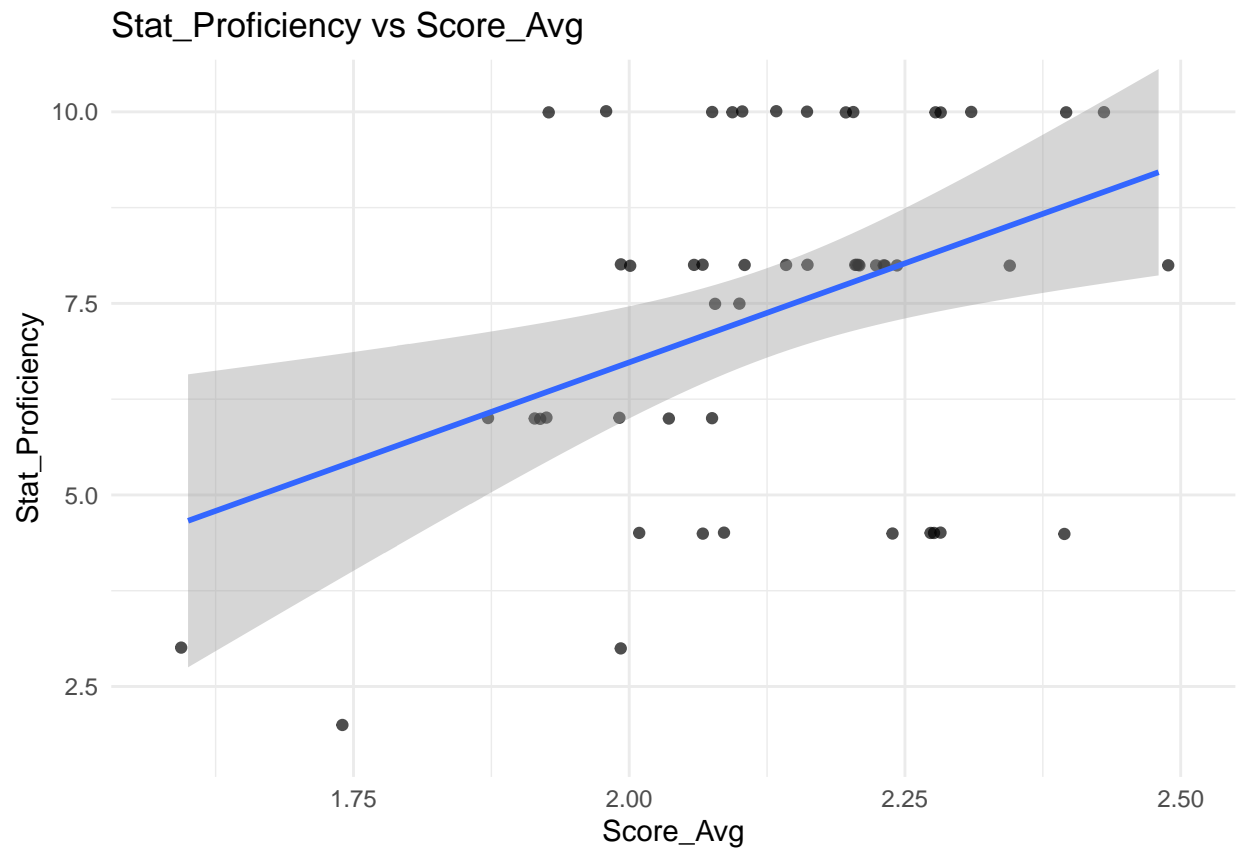
```
print(plots[[1]])
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



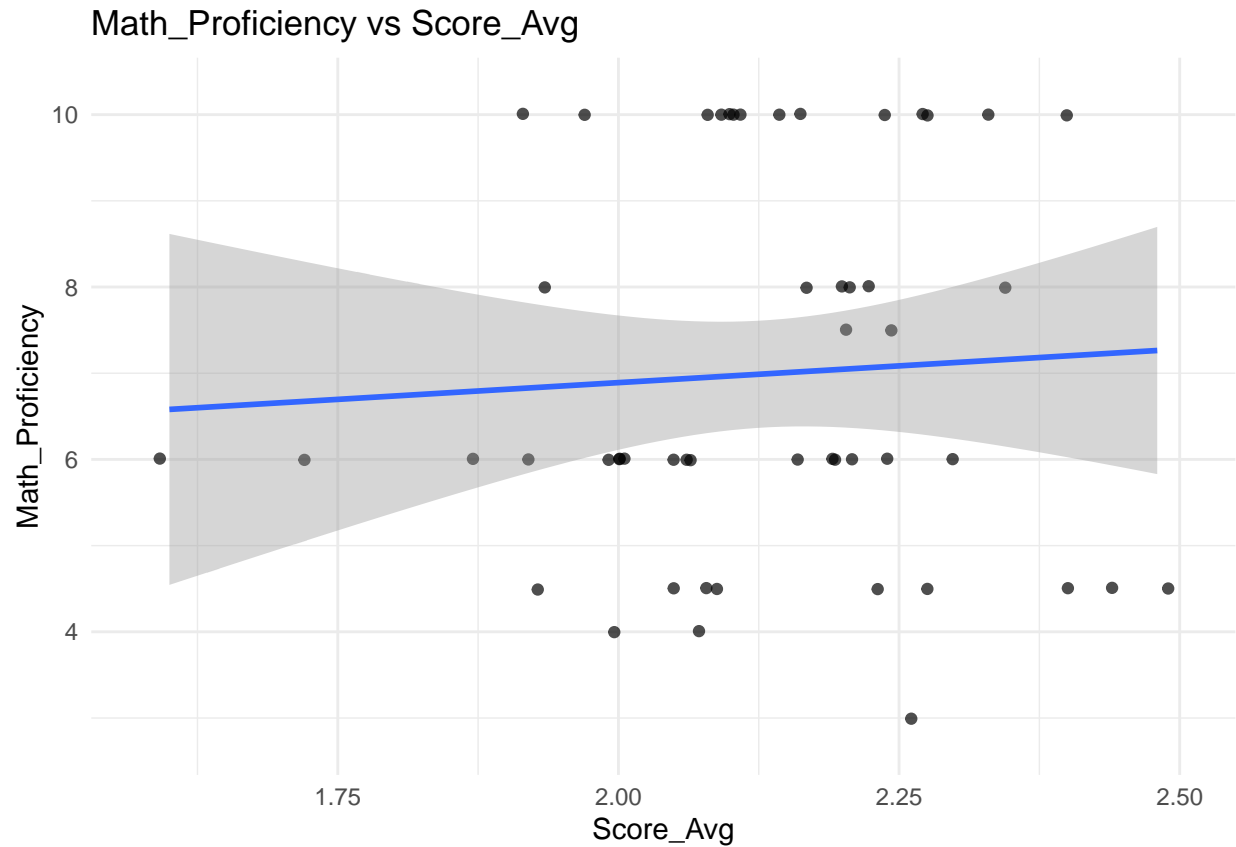
```
print(plots[[2]])
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



```
print(plots[[3]])
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



```
ggsave("results/figures/Prog_Proficiency_vs_Score_Avg.png", plots[[1]], width = 6, height = 4, dpi = 160)
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```

```
ggsave("results/figures/Stat_Proficiency_vs_Score_Avg.png", plots[[2]], width = 6, height = 4, dpi = 160)
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```

```
ggsave("results/figures/Math_Proficiency_vs_Score_Avg.png", plots[[3]], width = 6, height = 4, dpi = 160)
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```