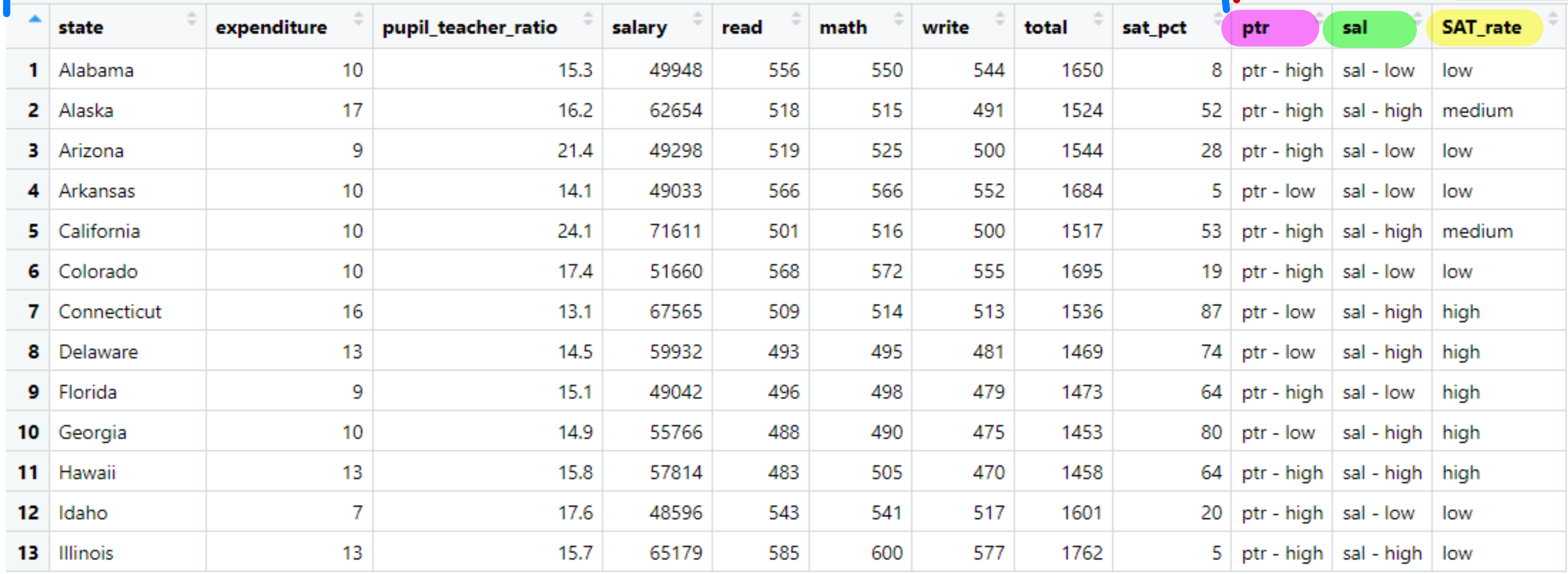
Data Visualization
Part 2

SAT_2010 dataset

Data

I mutated these to create categorical variables



	state	expenditure	pupil_teacher_ratio	salary	read	math	write	total	sat_pct	ptr	sal	SAT_rate
1	Alabama	10	15.3	49948	556	550	544	1650	8	ptr - high	sal - low	low
2	Alaska	17	16.2	62654	518	515	491	1524	52	ptr - high	sal - high	medium
3	Arizona	9	21.4	49298	519	525	500	1544	28	ptr - high	sal - low	low
4	Arkansas	10	14.1	49033	566	566	552	1684	5	ptr - low	sal - low	low
5	California	10	24.1	71611	501	516	500	1517	53	ptr - high	sal - high	medium
6	Colorado	10	17.4	51660	568	572	555	1695	19	ptr - high	sal - low	low
7	Connecticut	16	13.1	67565	509	514	513	1536	87	ptr - low	sal - high	high
8	Delaware	13	14.5	59932	493	495	481	1469	74	ptr - low	sal - high	high
9	Florida	9	15.1	49042	496	498	479	1473	64	ptr - high	sal - low	high
10	Georgia	10	14.9	55766	488	490	475	1453	80	ptr - low	sal - high	high
11	Hawaii	13	15.8	57814	483	505	470	1458	64	ptr - high	sal - high	high
12	Idaho	7	17.6	48596	543	541	517	1601	20	ptr - high	sal - low	low
13	Illinois	13	15.7	65179	585	600	577	1762	5	ptr - high	sal - high	low

We added three categorical variables to the dataset (SAT_2010):

```
SAT_2010 <- SAT_2010 %>%
 mutate(ptr = ifelse(pupil_teacher_ratio >= 15, "ptr - high","ptr -
 low"),
       sal = ifelse(salary >= 52000, "sal - high","sal - low"),
       SAT_rate = cut(
         sat_pct,
         breaks = c(0, 30, 60, 100),
         labels = c("low", "medium", "high")
 ))
```

← new function used in mutate to create categories. The variable it creates is factor.

Uses this column to create categories ← sat_pct

creates intervals:

(0,30] ← "low"

(30,60] ← "medium"

(60,100] ← "high"

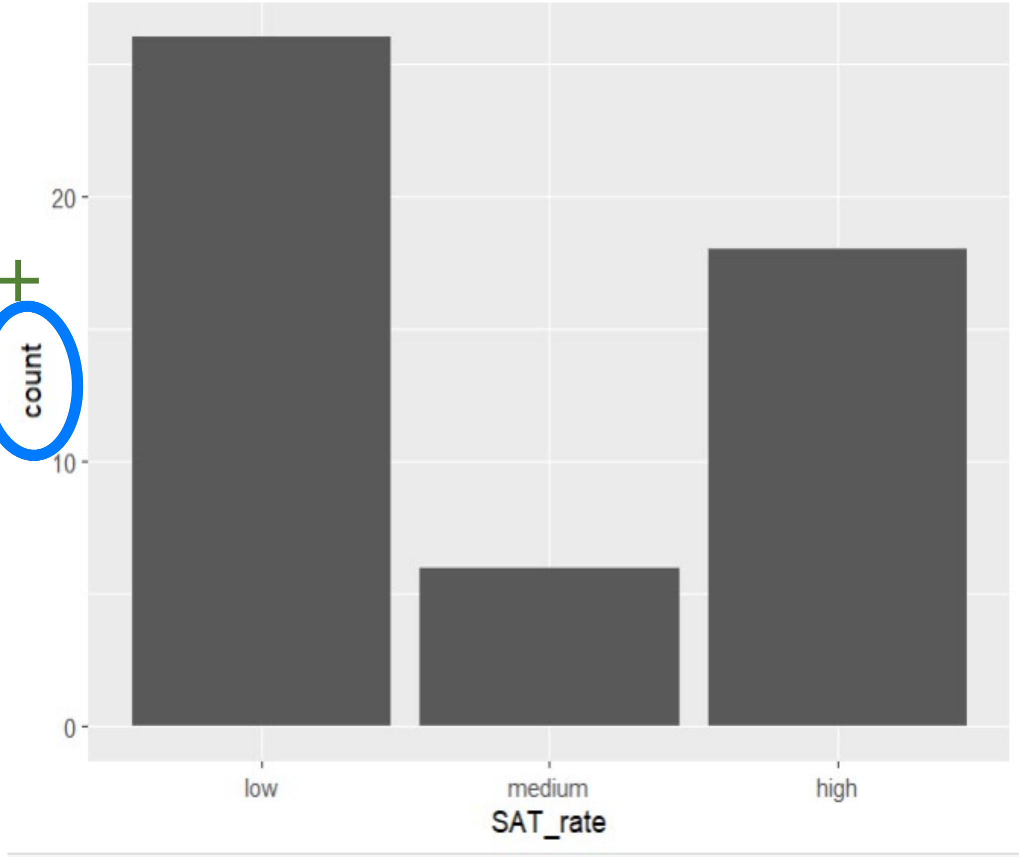
Notice 0 is not part of the interval.

Multivariate Displays
Bar Graphs

Figure 1: Make a bar graph with SAT_rate. Notice that this is a One Variable Bar Graph (it counts by default)

One variable bar graph

```
fig_1 <- SAT_2010 %>%
  ggplot(aes(x = SAT_rate)) +
  geom_bar()
fig_1
```



counts by default

Figure 2: Make a bar graph with SAT_rate on the x axis and The Average Math Score on the y axis. Notice that this is a two Variable Bar Graph (you need stat = "identity")

```
fig_2 <- SAT_2010 %>%
  ggplot(aes(x = SAT_rate,
             y = math)) +
  geom_bar(stat = "identity")
fig_2
```

allows you to have a variable y in a bar graph.

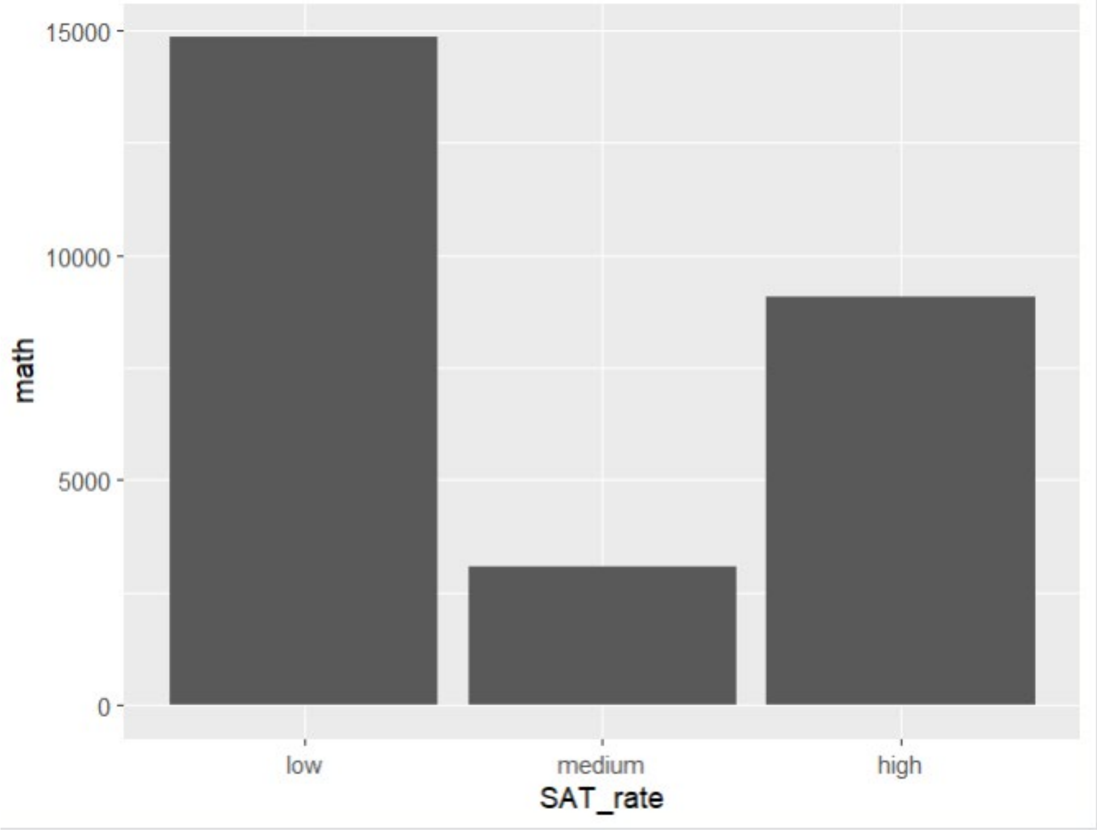


Figure 3: Make a bar graph with SAT_rate on the x axis and The Average Math Score on the y axis. Make every column a different color.

```
fig_3 <- SAT_2010 %>%
  ggplot(aes(x = SAT_rate,
             y = math,
              fill = SAT_rate)) +
  geom_bar(stat = "identity")
fig_3
```

splits each category into a different color

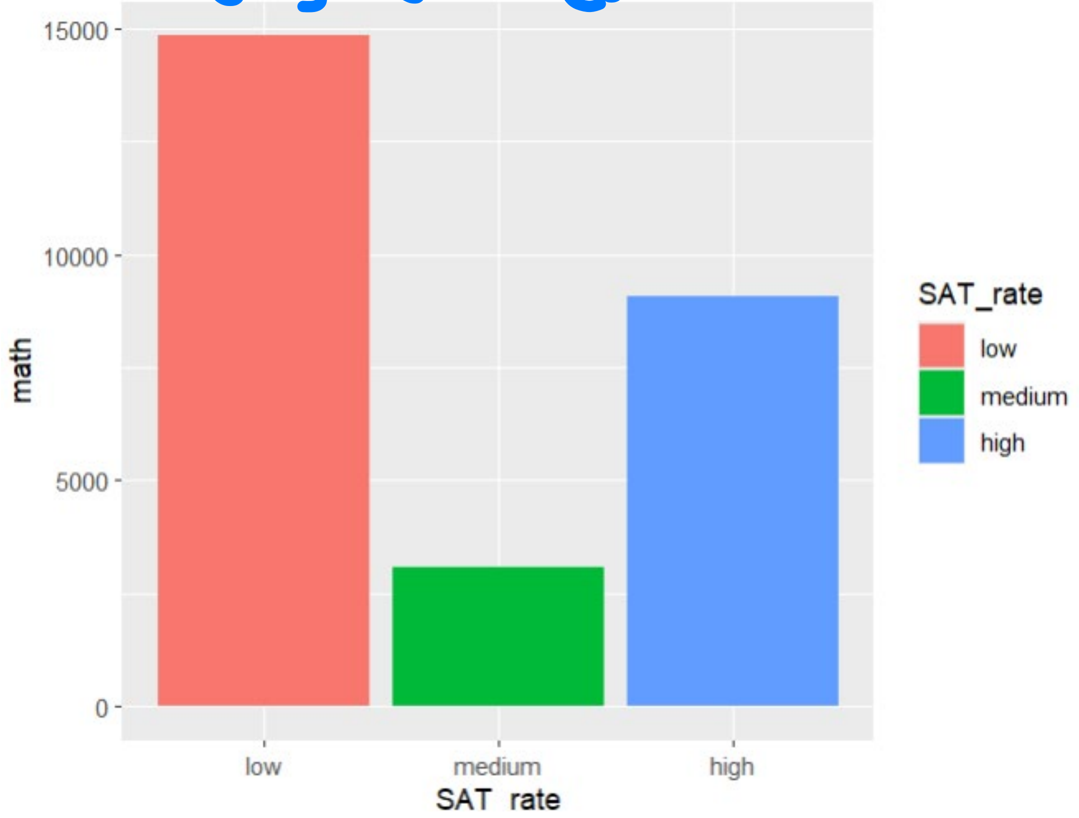


Figure 4: (Stacking by a third variable). Make a bar graph with SAT_rate on the x axis and The Average Math Score on the y axis. Stack the bars by the variable ptr.

These two variable are different.

```
fig_4 <- SAT_2010 %>%
  ggplot(aes(x = SAT_rate,
             y = math,
               fill = ptr)) +
  geom_bar(stat = "identity")
fig_4
```

should be categorical

This creates a stacked bar graph

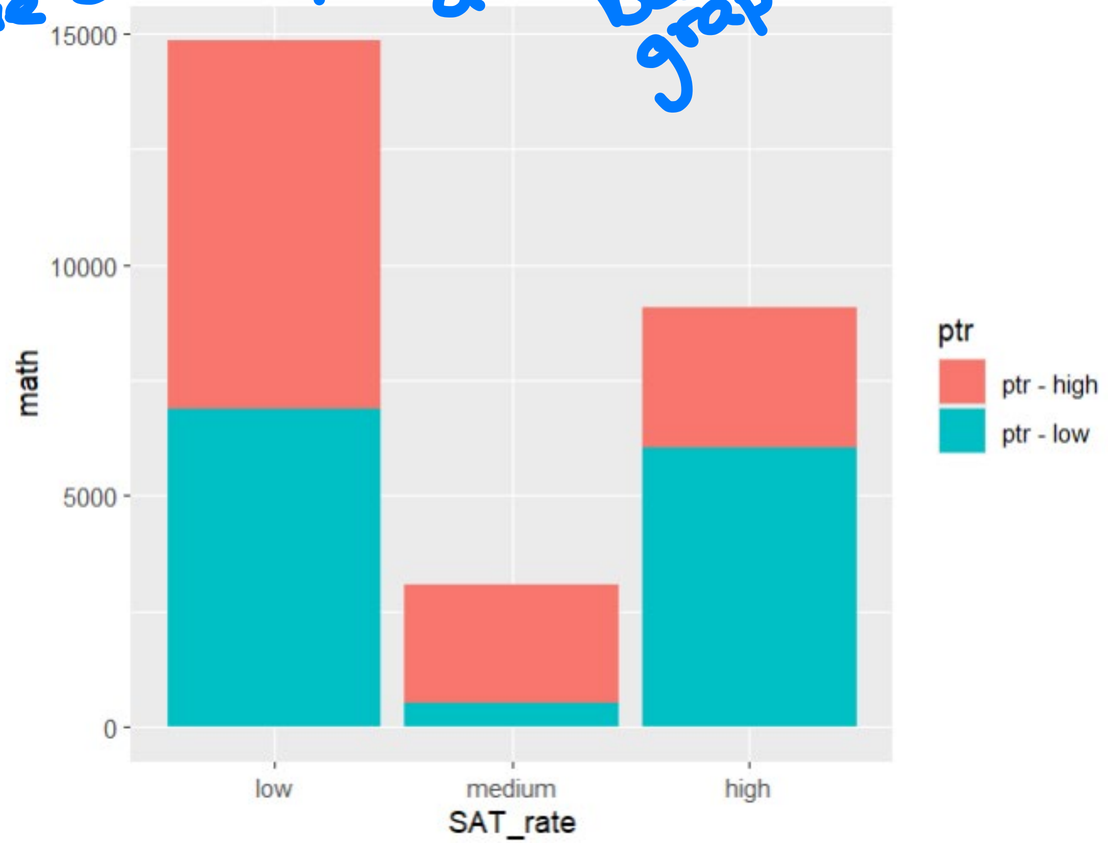
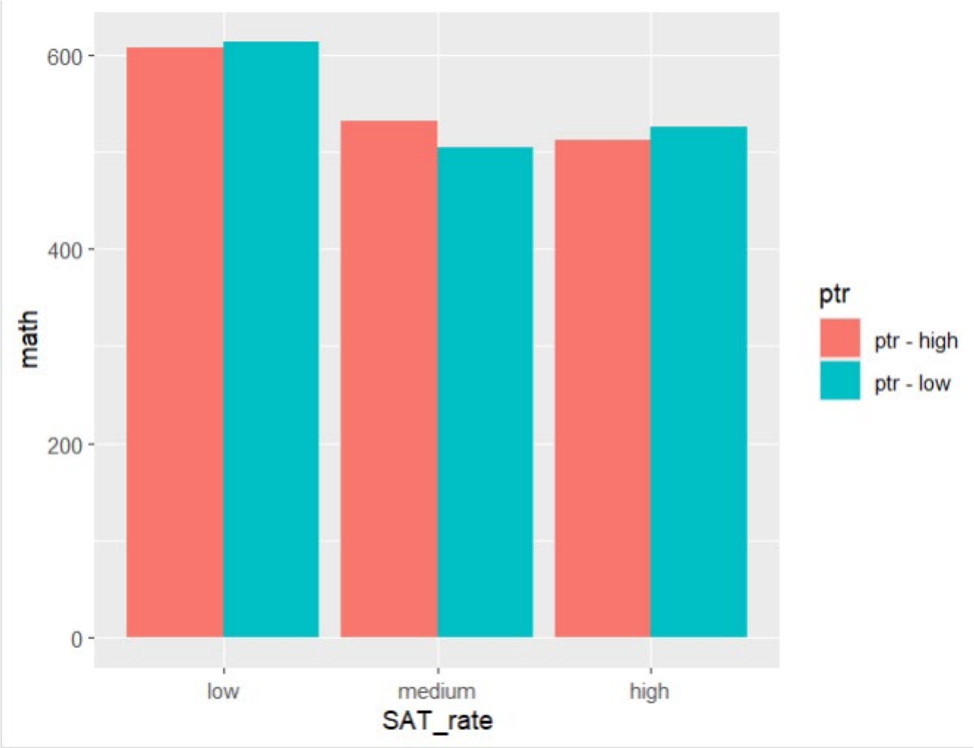


Figure 5: (Grouped Bar Graph). Make a bar graph with SAT_rate on the x axis and The Average Math Score on the y axis. Group the bars by the variable ptr.

```
fig_5 <- SAT_2010 %>%
  ggplot(aes(x = SAT_rate,
             y = math,
             fill = ptr)) +
  geom_bar(stat = "identity",
           position = "dodge")
fig_5
```

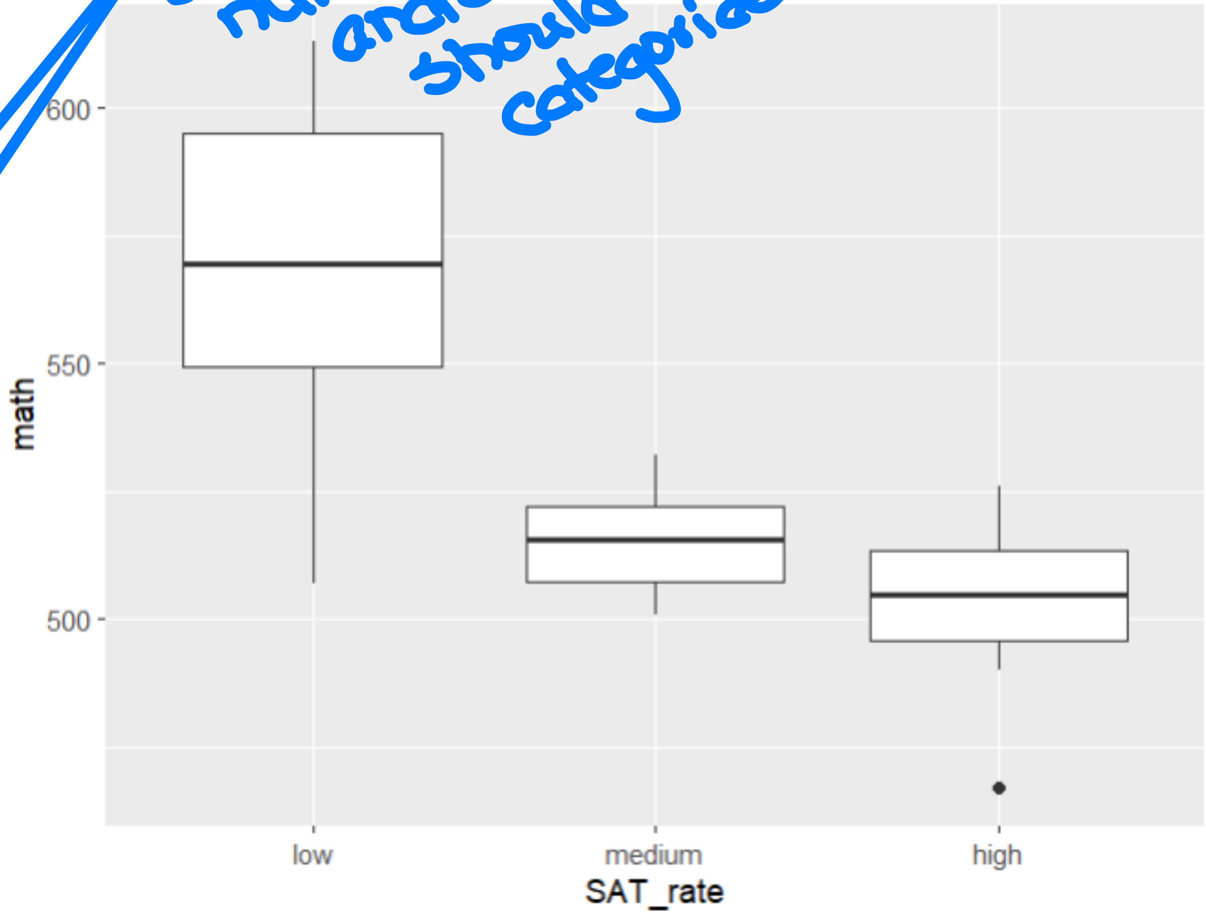
This changes the bar graph from "stacked to "grouped "



Box Plots

Figure 6: Make side by side box plots with SAT_rate on the x axis and The Average Math Score on the y axis.

```
fig_6 <- SAT_2010 %>%
  ggplot(aes(x = SAT_rate,
             y = math)) +
  geom_boxplot()
fig_6
```



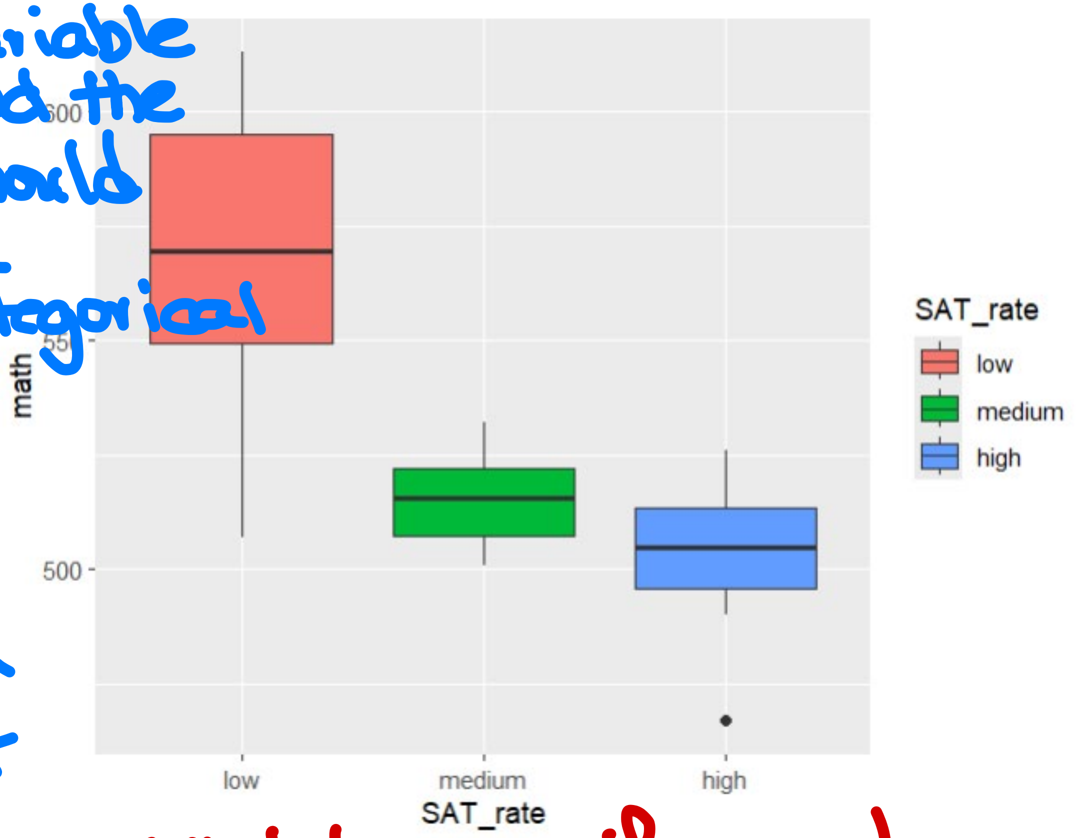
one should be numerical and another should be categorical

Figure 7: Make side by side box plots with SAT_rate on the x axis and The Average Math Score on the y axis. Change the filling color of each SAT_rate.

```
fig_7 <- SAT_2010 %>%
  ggplot(aes(x = SAT_rate,
             y = math,
             fill = SAT_rate)) +
  geom_boxplot()
fig_7
```

Same variable and the should be categorical

gives each category a different color.



What happens if you do color = SAT_rate? Try it!

Scatter Plots

Figure 8: Create a Scatter Plot on the Expenditure and The Average Math SAT Score.

In a Scatter plot both x and y should be numerical

```
fig_8 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math)) +
  geom_point()
fig_8
```

Scatter plot

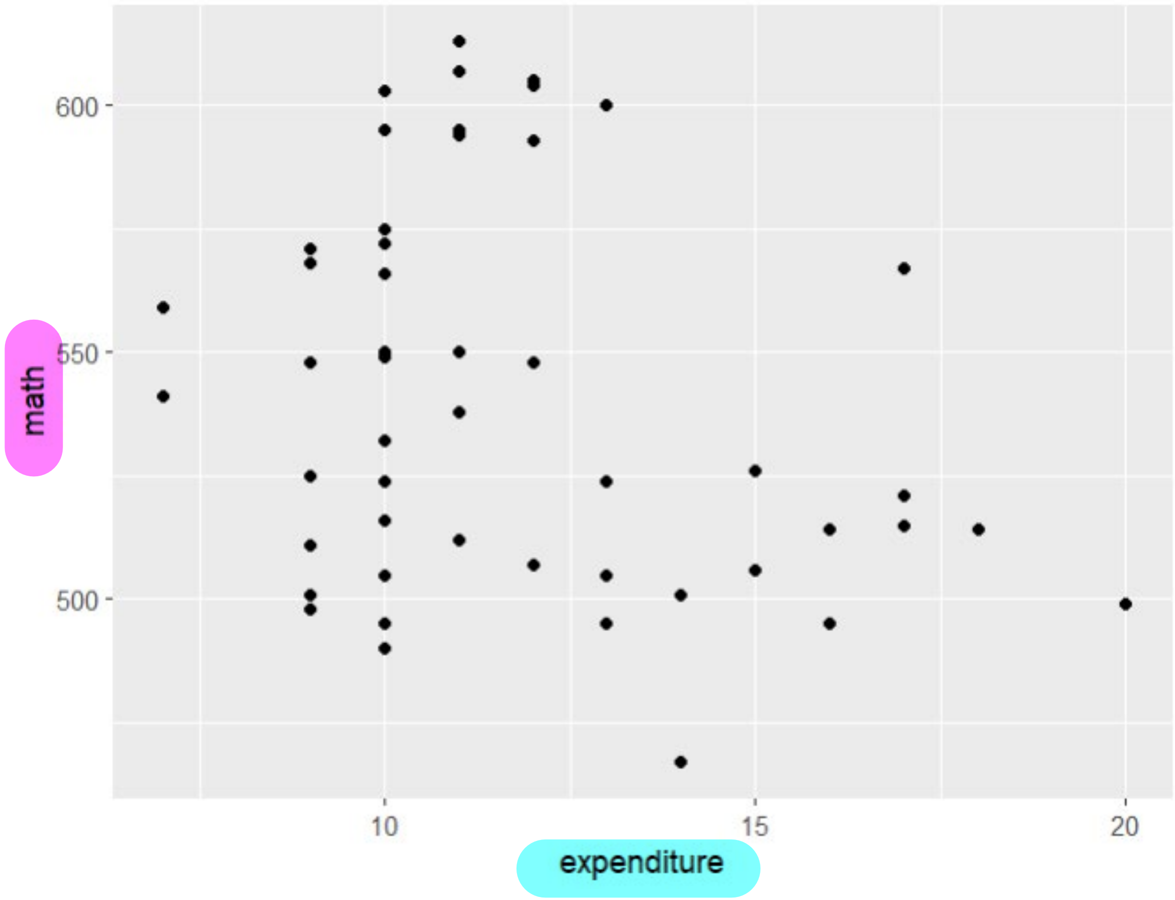


Figure 9: Create a Scatter Plot on the Expenditure and The Average Math SAT Score and add a trend line with ggplot.

```
fig_9 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math)) +
  geom_point() +
  geom_smooth(method = "lm",
              se = FALSE)

fig_9
```

linear model

"fits a line or a polynomial on the data"

error band

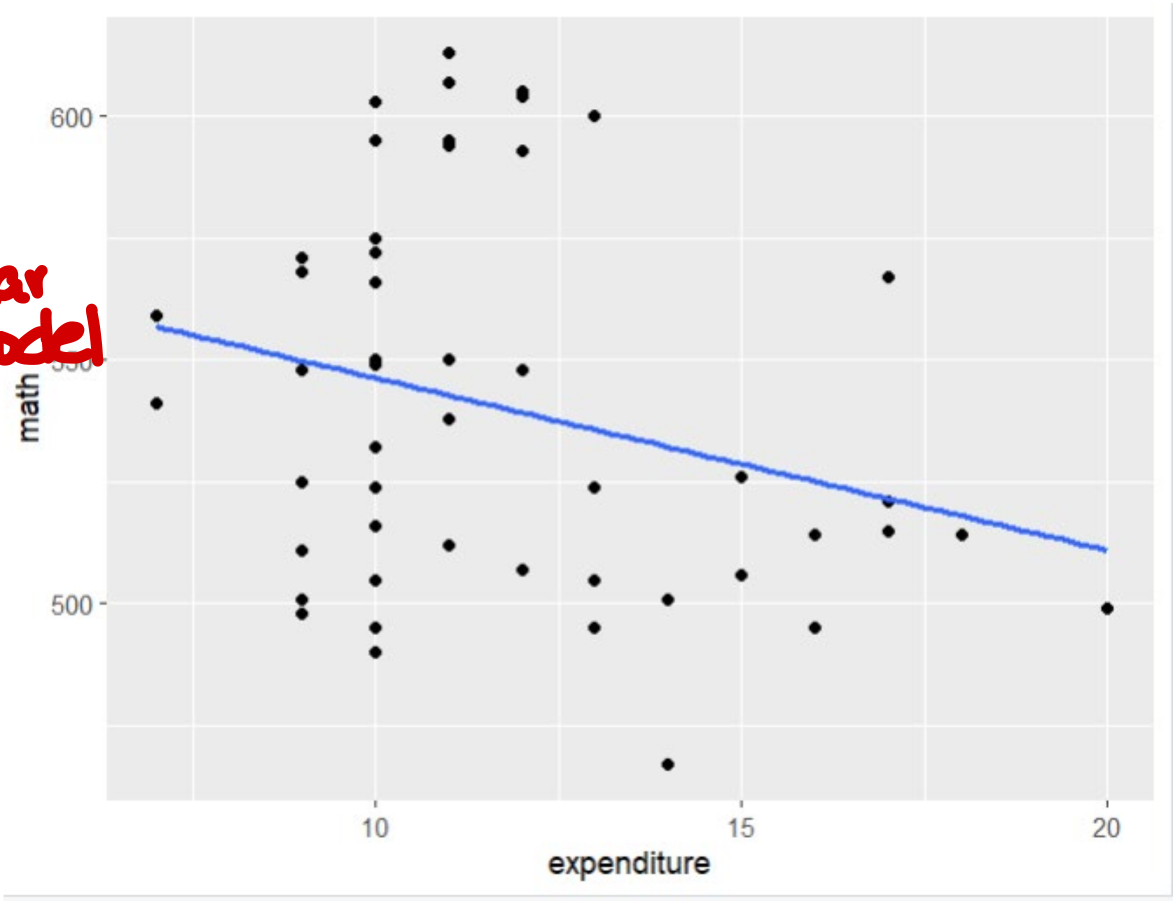
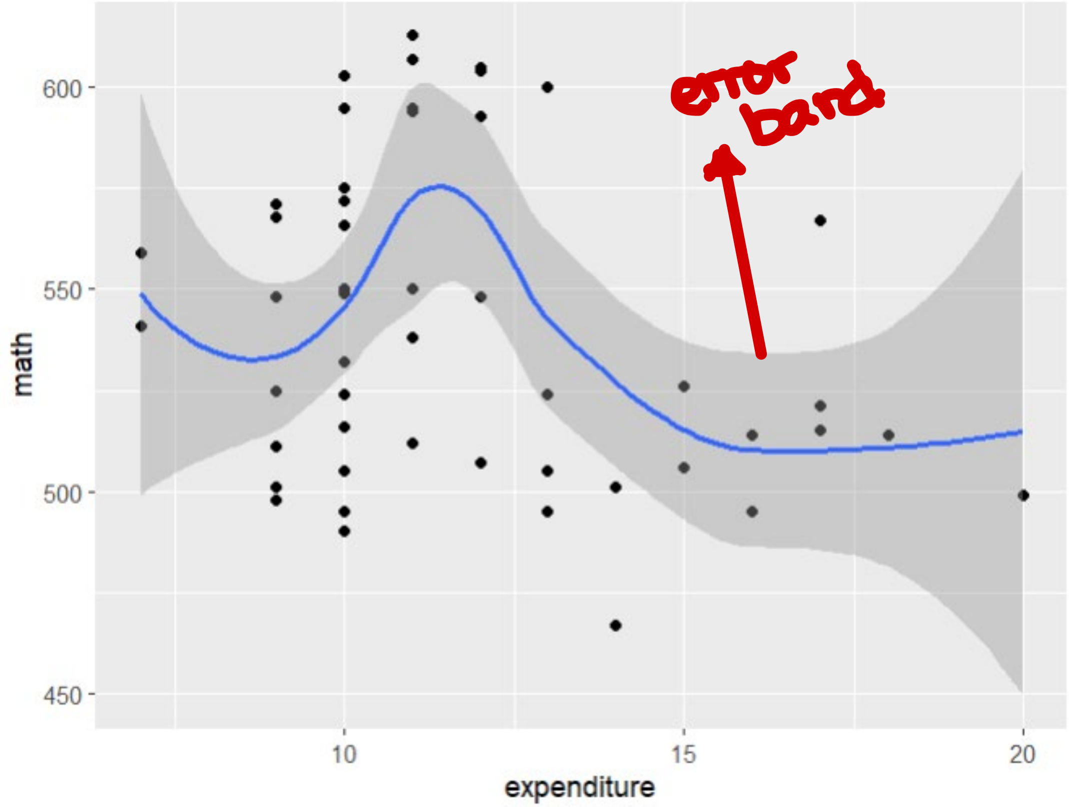


Figure 10: Create a Scatter Plot on the Expenditure and The Average Math SAT Score and add a polynomial fitting with the standard error band.

```
fig_10 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math)) +
  geom_point() +
  geom_smooth(method = "loess",
              se = TRUE)
fig_10
```

fits a polynomial



error band

Figure 11: Create a Scatter Plot on the Expenditure and The Average Math SAT Score and split the data with different colors by SAT_rate.

numerical

categorical

```
fig_11 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math,
             color = SAT_rate)) +
  geom_point()
fig_11
```

splits data by categories in that variable

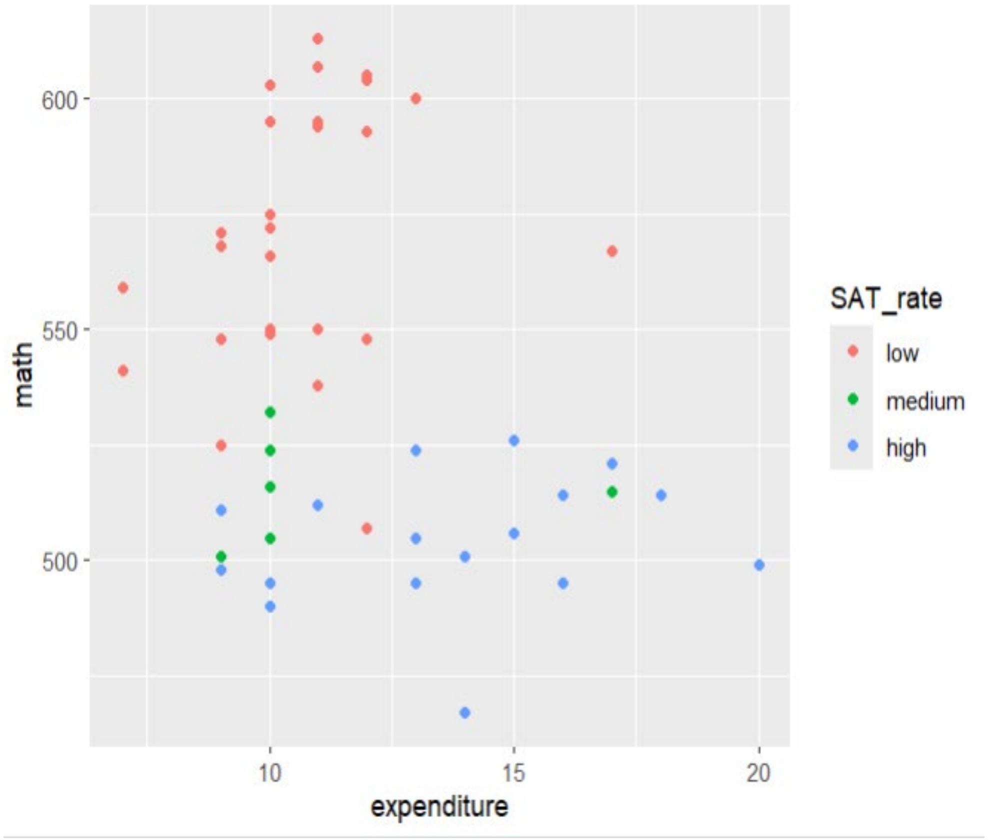
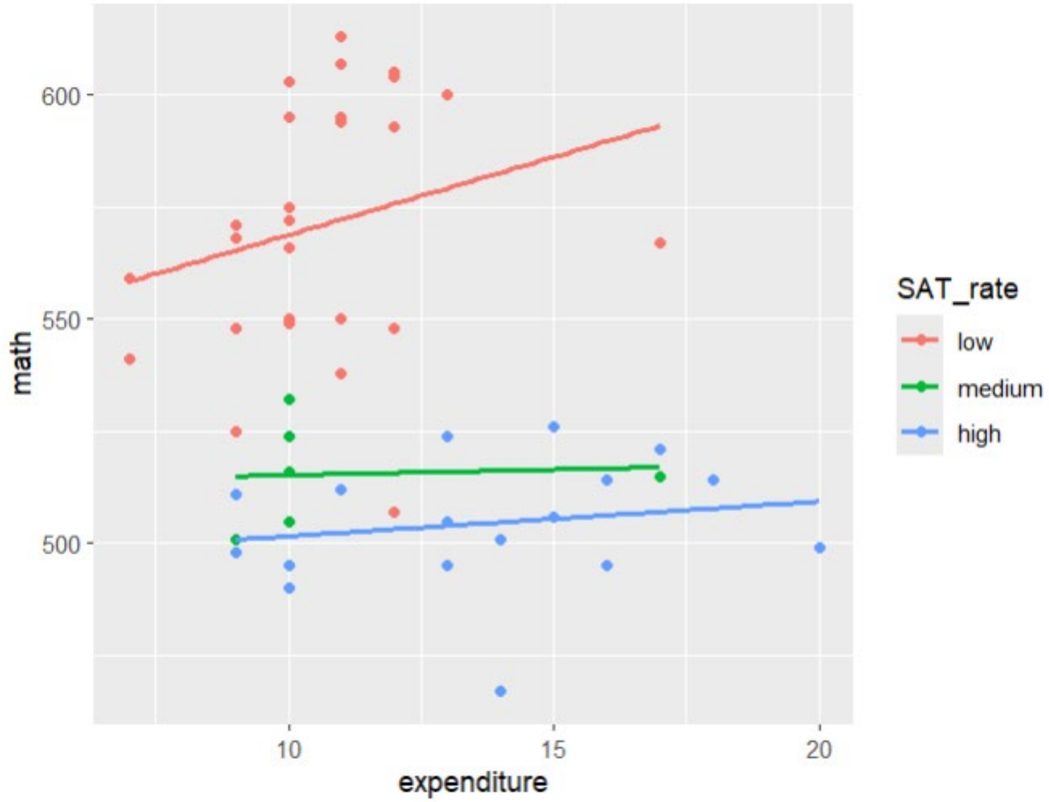


Figure 12: Create a Scatter Plot on the Expenditure and The Average Math SAT Score and split the data with different colors by SAT_rate. Add a trend line.

```
fig_12 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math,
             color = SAT_rate)) +
  geom_point() +
  geom_smooth(method = "lm",
              se = FALSE)
fig_12
```



Fits a line for each category

Figure 13: Create a Scatter Plot on the Expenditure and The Average Math SAT Score and split the data with different shapes by SAT_rate.

```
fig_13 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math,
             shape = SAT_rate)) +
  geom_point()
fig_13
```

splits by different shapes

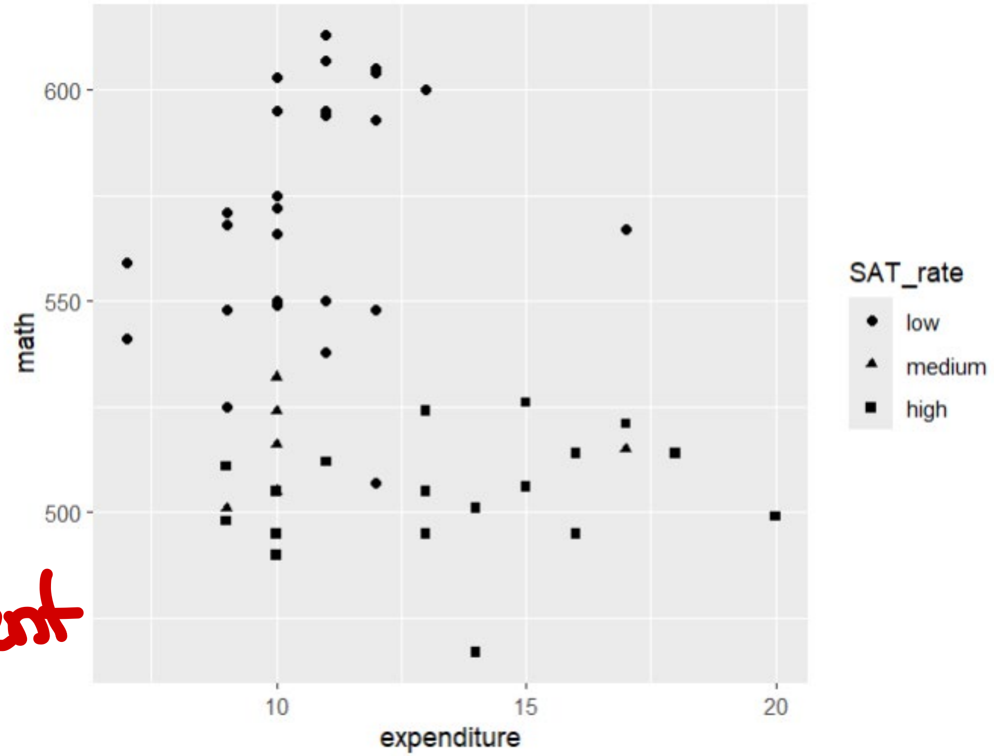
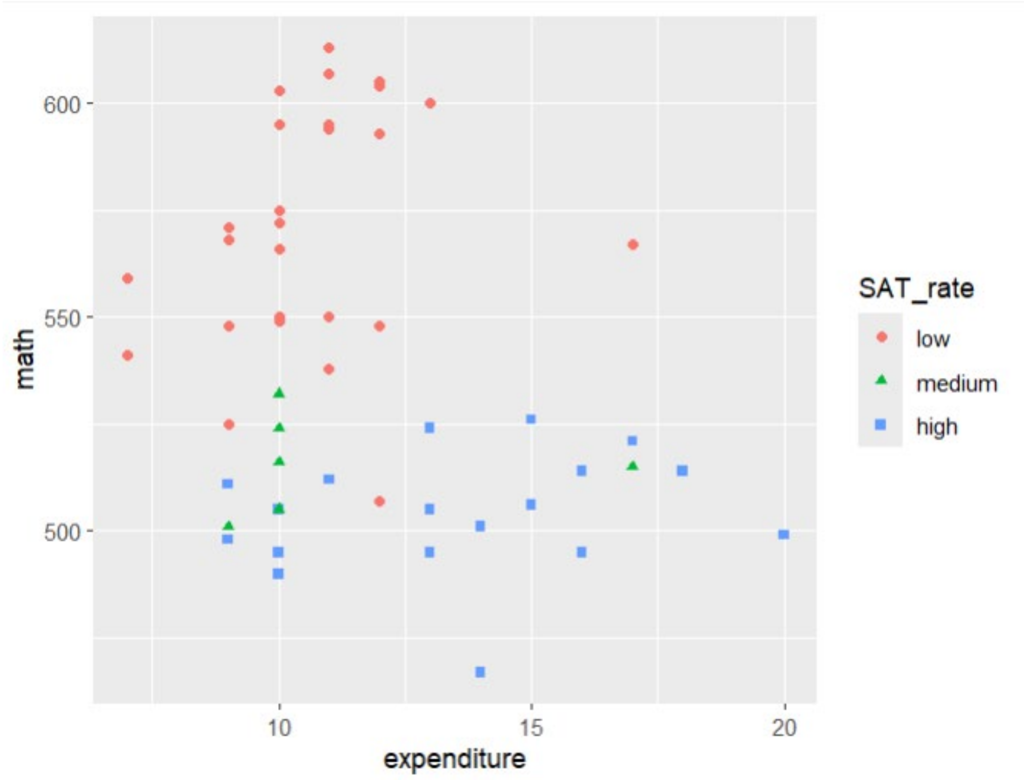


Figure 14: Create a Scatter Plot on the Expenditure and The Average Math SAT Score and split the data with different shapes and colors by SAT_rate.

```
fig_14 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math,
             color = SAT_rate,
             shape = SAT_rate)) +
  geom_point()
fig_14
```

Splits by shapes and colors.



Faceting

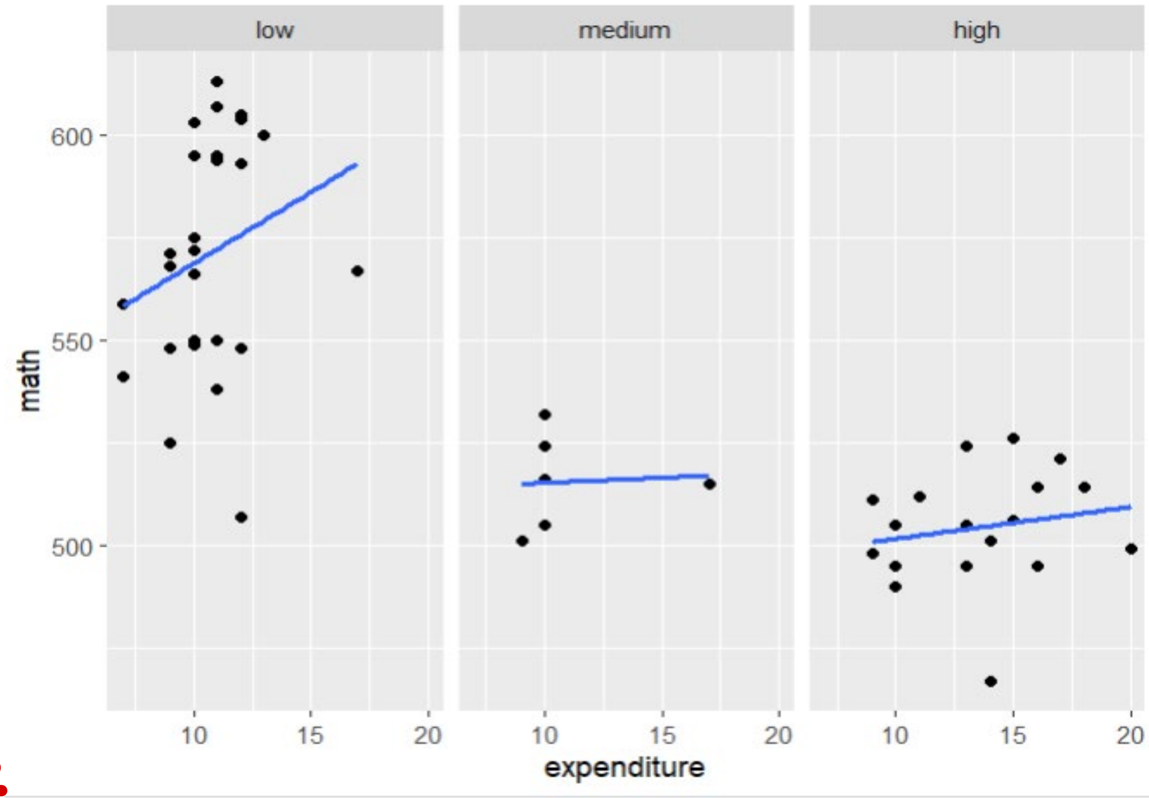
This is similar to using shape or color for a categorical variables but puts them on separate plots.

Facet with 1 variable

Figure 15: Create a Scatter Plot on the Expenditure and The Average Math SAT Score and split the data by SAT_rate where each category is a separate plot. Include a trend line.

```
fig_15 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math)) +
  geom_point() +
  geom_smooth(method = "lm",
              se = FALSE) +
  facet_wrap(~SAT_rate)
fig_15
```

splits by one variable



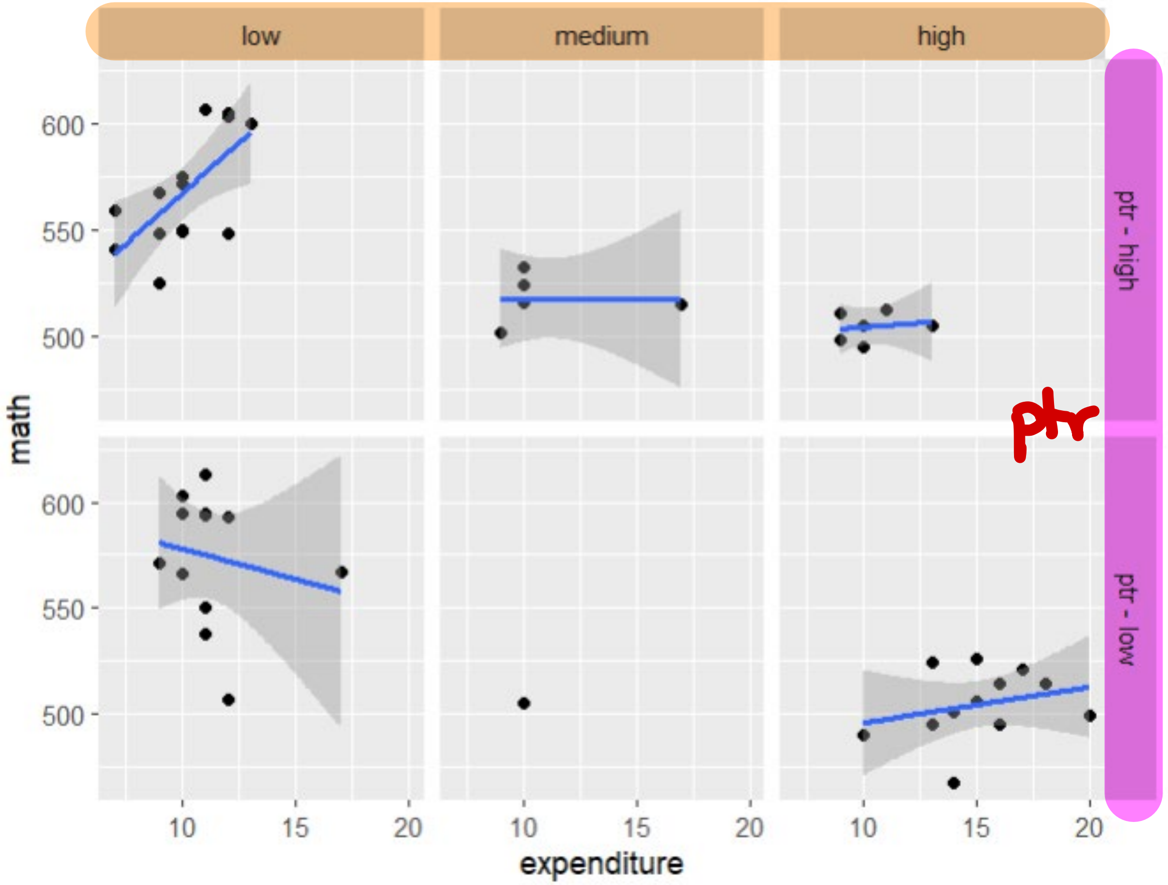
Facet with 2 variables

Figure 16: Create a Scatter Plot on the Expenditure and The Average Math SAT Score and split the data by SAT_rate and ptr where each pair of categories is a separate plot. Include a trend line with a standard error band.

```
fig_16 <- SAT_2010 %>%
  ggplot(aes(x = expenditure,
             y = math)) +
  geom_point() +
  geom_smooth(method = "lm",
              se = TRUE) +
  facet_grid(ptr ~ SAT_rate)
fig_16
```

splits by TWO variables

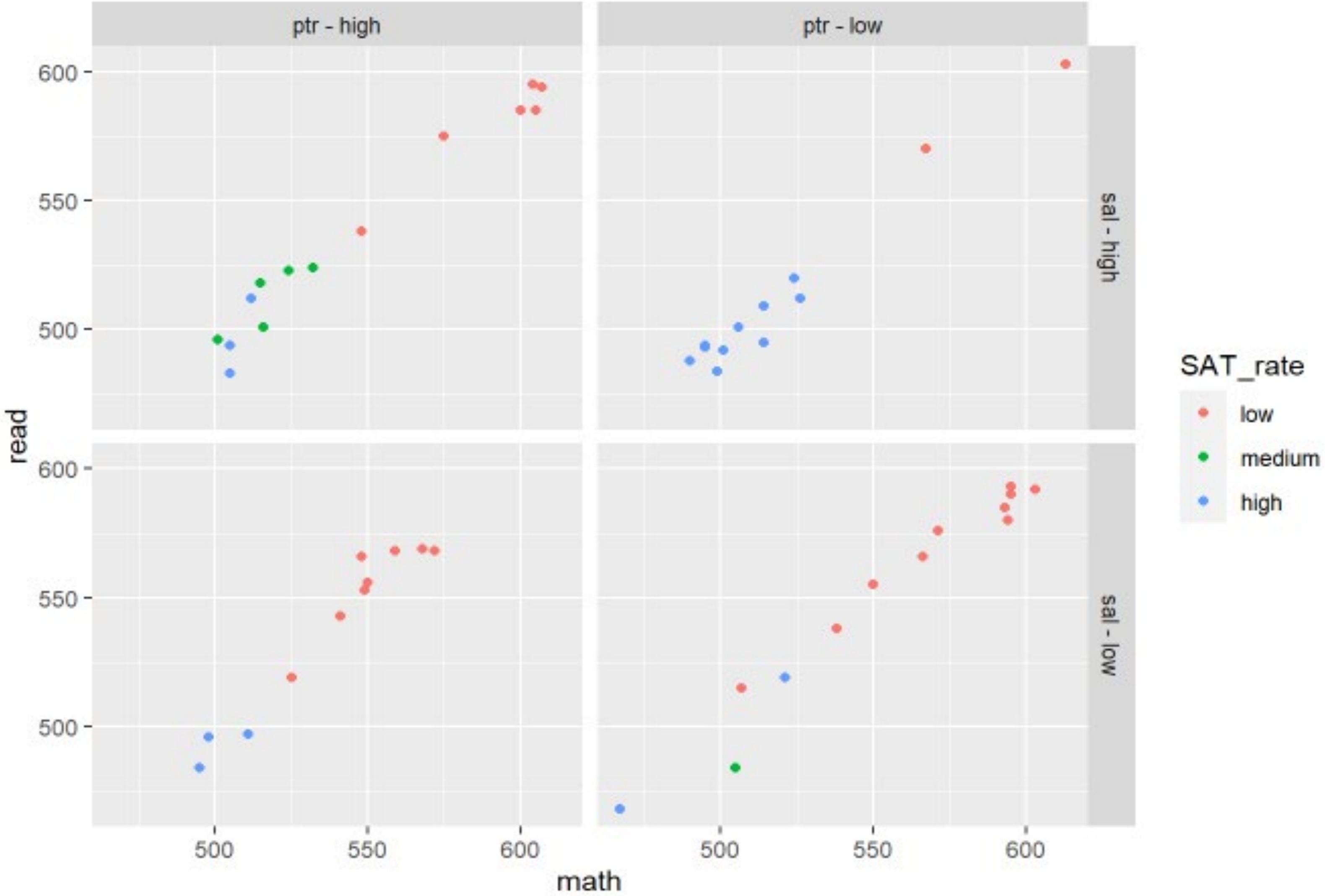
SAT_rate



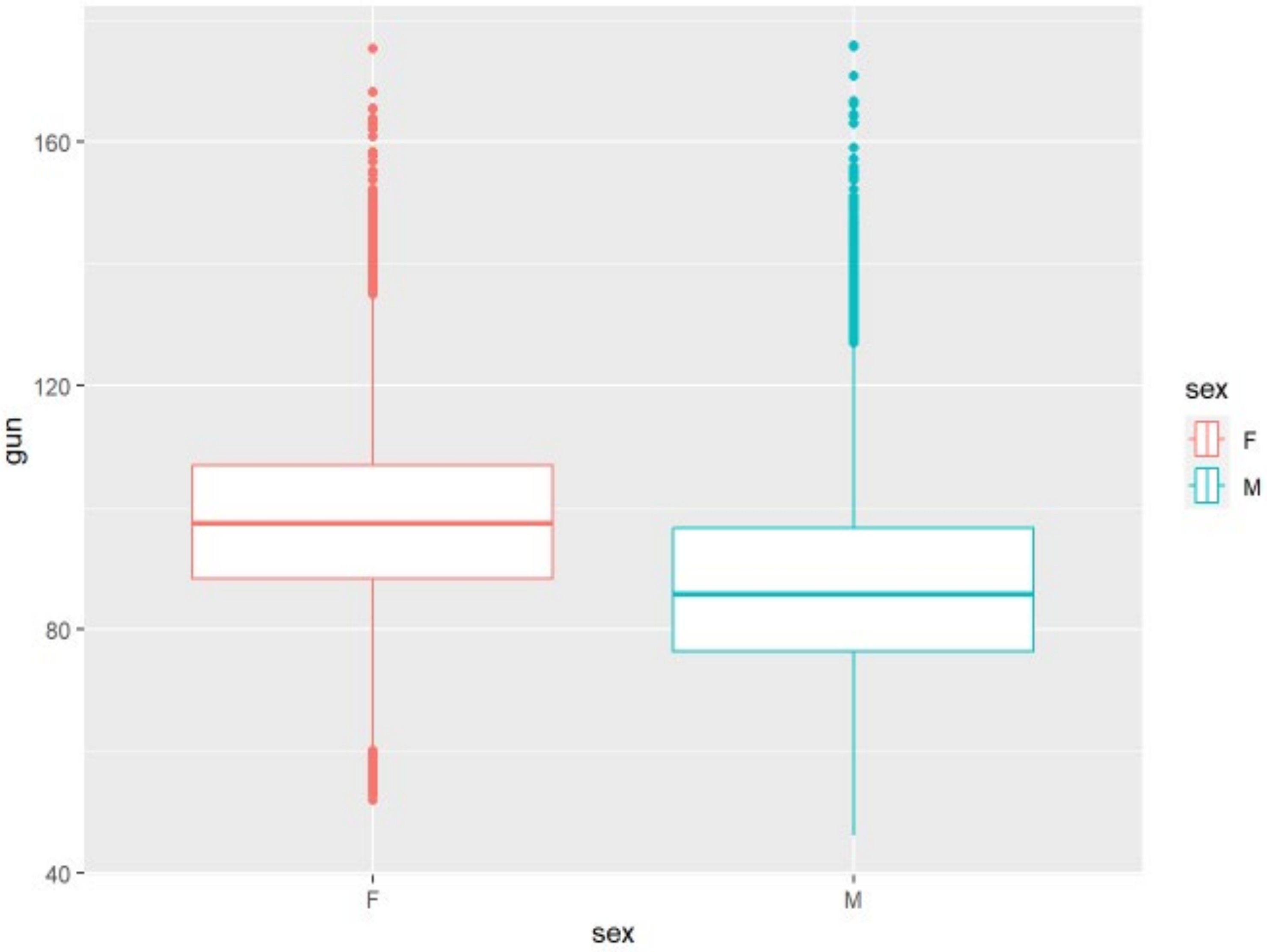
ptr

Do it Yourself –
Recreate These Plots

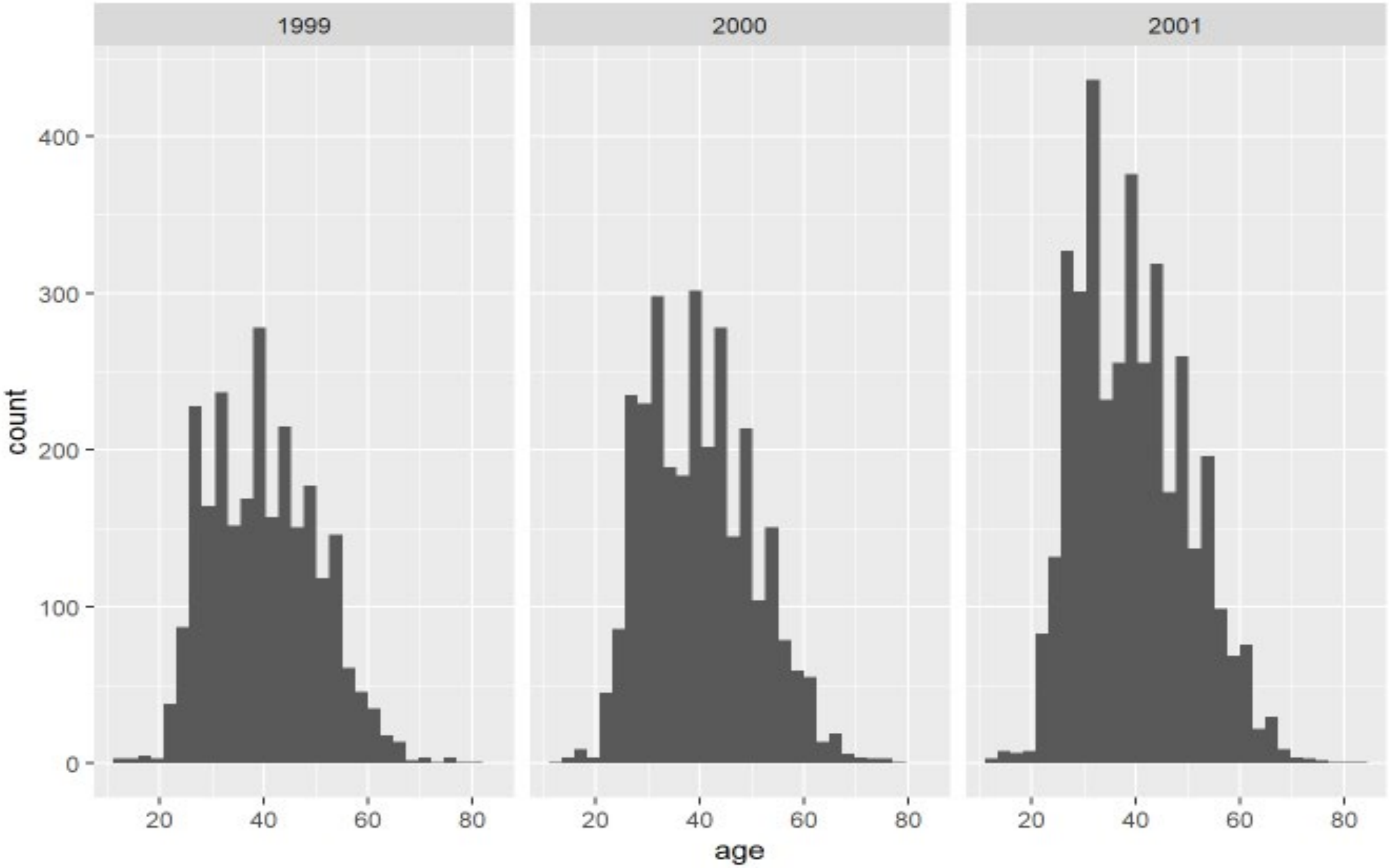
Plot # 1


ptr - high
ptr - low
sal - high
sal - low
read
600
550
500
600
550
500
math
500
550
600
500
550
600
SAT_rate
low
medium
high

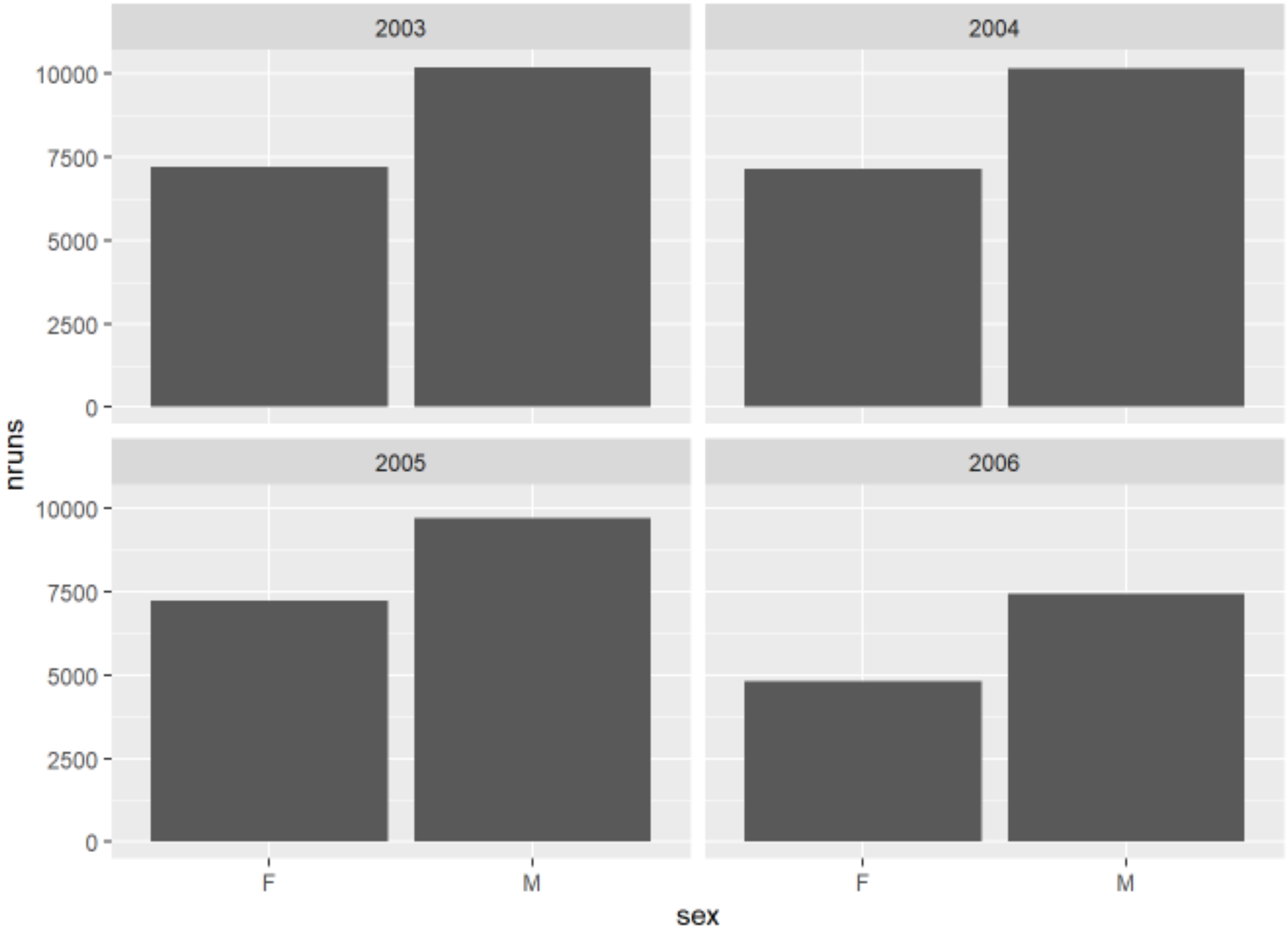
Plot # 2



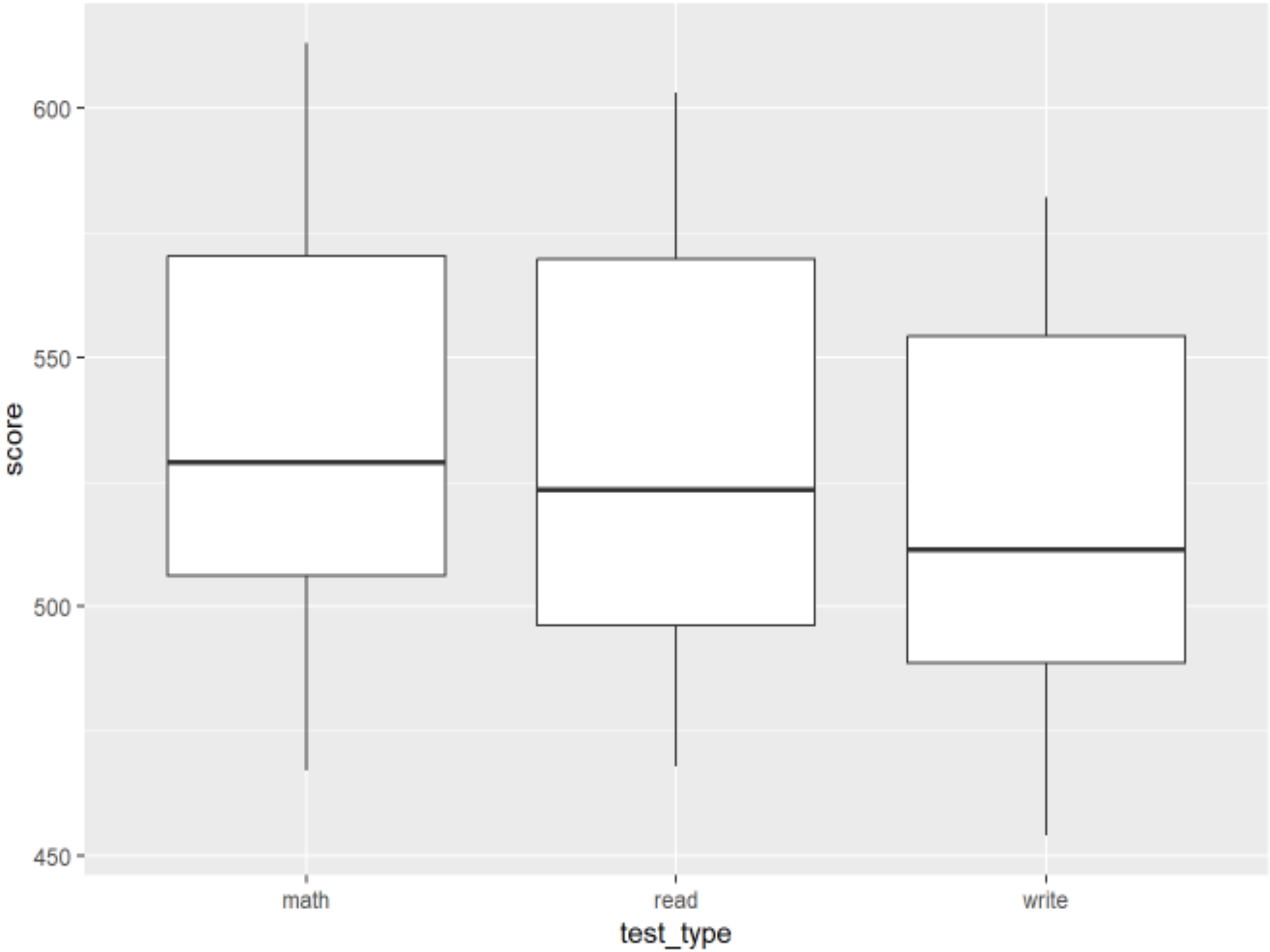
Plot # 3



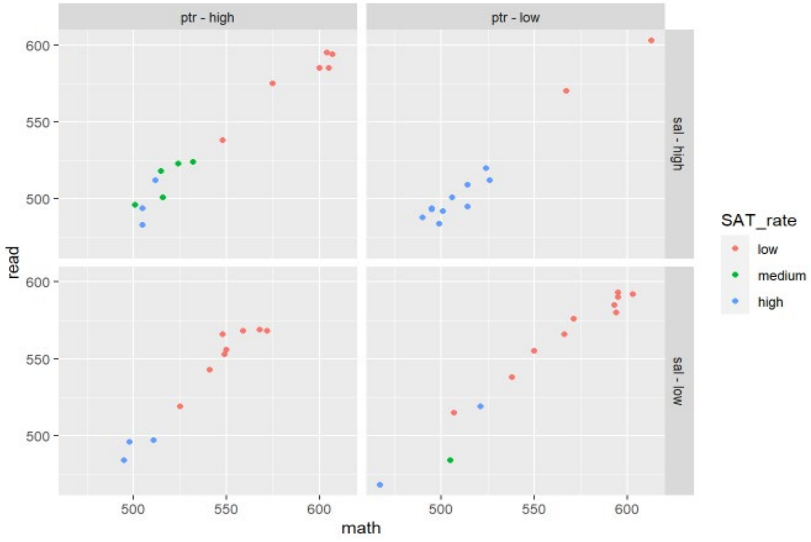
Plot # 4



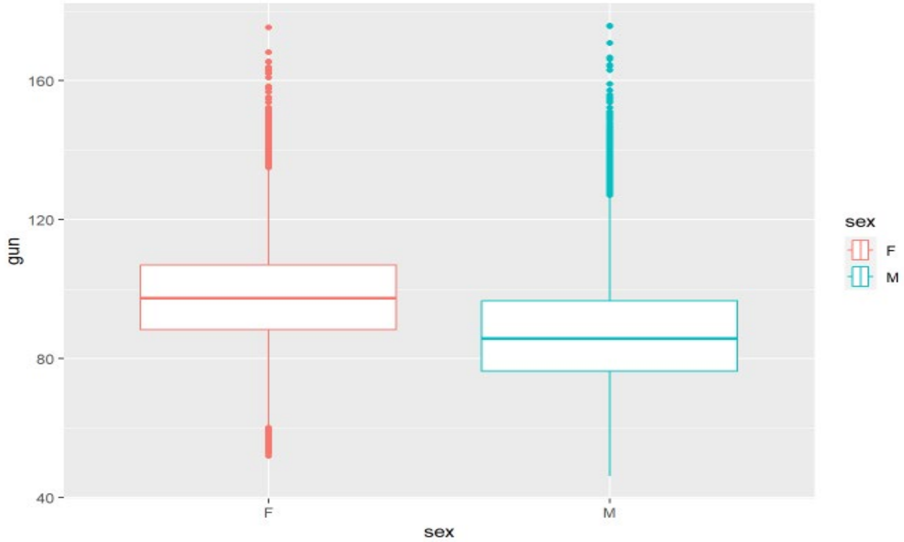
Plot # 5



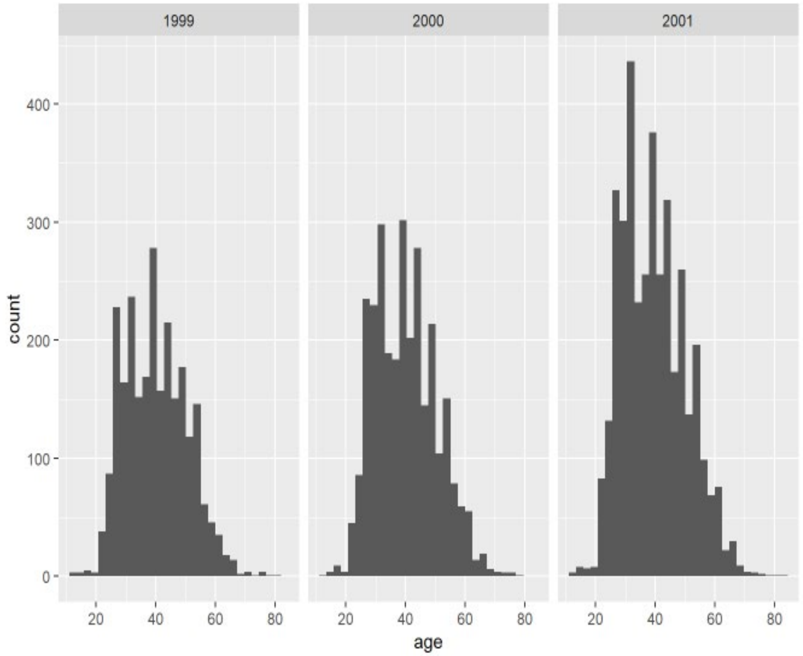
Plot # 1



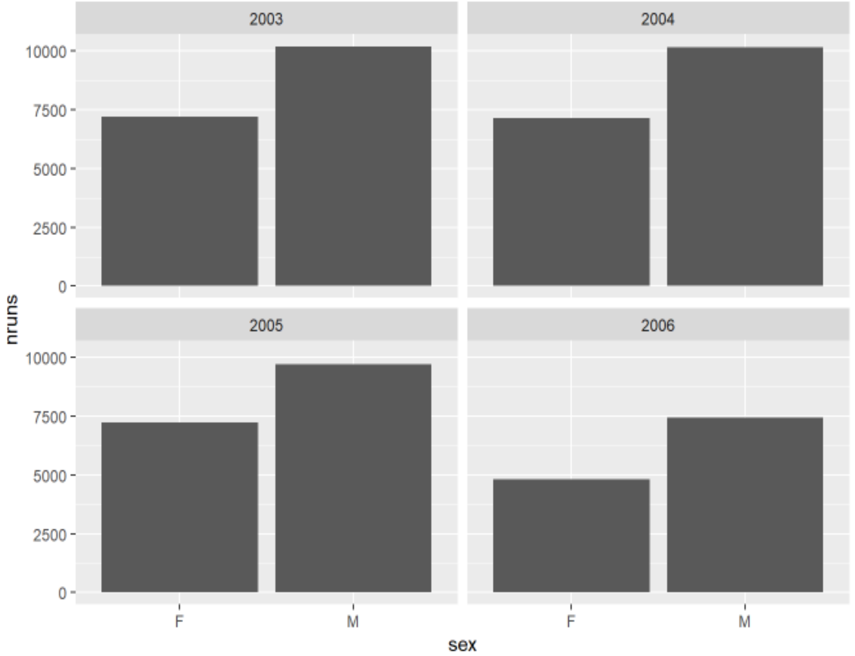
Plot # 2



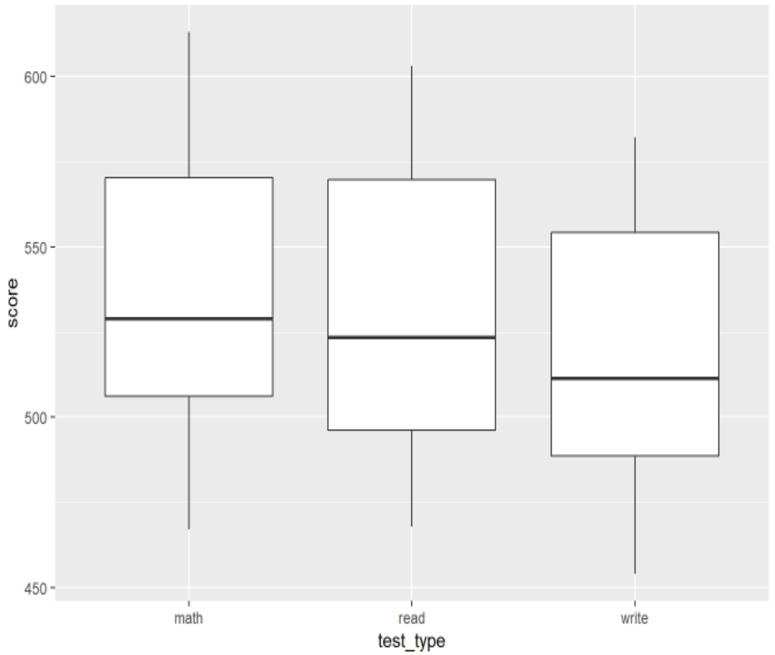
Plot # 3



Plot # 4



Plot # 5


score
test_type
math
read
write