CA Programming Data Project

Isaac Paulson

This analysis focuses on the claim that iReady's assessment has been proven to correlate with many states' end-of-year summative assessments. The analysis uses fabricated data from hypothetical state test and i-Ready. The analysis involves combining the two data sets, removing invalid data, determining proficiency levels for the i-Ready test that can be compared to the proficiency levels of the state test, and calculating the percent of students at the proficient level for both the state test and i-Ready.

```
# check for necessary packages and install if not installed
list.of.packages <- c("dplyr", "ggplot2", "knitr", "magrittr", "readxl", "tidyr")
new.packages <- list.of.packages[!(list.of.packages %in% installed.packages()[,"Package"])]
if(length(new.packages)) install.packages(new.packages)

# attach necessary packages
library(dplyr)
library(ggplot2)
library(knitr)
library(magrittr)
library(readxl)
library(tidyr)
```

Data Preparation

After checking for and installing the necessary packages, load the provided data files into R. For some students, the files contain lettered codes instead of scores. These codes — E and M from the state test, P and X from i-Ready — will be loaded as NA to make it easier to work with data types and eliminate invalid scores from the analysis later on. View the `summary()` for i-Ready scores.

```
data_file <- file.path(".", "Programming Project Data.xlsx")

no_score_codes = c("P","X","E","M")

state_test_scores <- read_excel(data_file, "State Test Scores", na=no_score_codes)
iReady_scores <- read_excel(data_file, "i-Ready Scores", na=no_score_codes)

summary(iReady_scores)
```

```
##     User_ID          School          Student_Grade   iReady_Subject
##  Min.   :100292   Length:2790        Min.   :2.000   Length:2790
##  1st Qu.:315255   Class :character   1st Qu.:4.000   Class :character
##  Median :550539   Mode  :character   Median :6.000   Mode  :character
##  Mean   :549047                      Mean   :5.509
##  3rd Qu.:776441                      3rd Qu.:7.000
##  Max.   :999918                      Max.   :9.000
##
##   iReady_Score   iReady_Placement
##  Min.   :  0.0   Length:2790
```

```
##  1st Qu.:541.8   Class :character
##  Median :583.0   Mode  :character
##  Mean   :580.8
##  3rd Qu.:624.2
##  Max.   :903.0
##  NA's   :2
```

View the `summary()` for state test scores.

```
summary(state_test_scores)
```

```
##     User_ID        Test_Name             Score       Proficiency
##  Min.   :100292   Length:2793        Min.   :100.0   Length:2793
##  1st Qu.:315017   Class :character   1st Qu.:366.0   Class :character
##  Median :550492   Mode  :character   Median :417.0   Mode  :character
##  Mean   :548856                      Mean   :417.7
##  3rd Qu.:776384                      3rd Qu.:465.0
##  Max.   :999918                      Max.   :999.0
##                                      NA's   :8
##     School          Student_Grade   iReady_Subject      iReady_Score
##  Length:2793        Min.   :2.000   Length:2793        Min.   :  0.0
##  Class :character   1st Qu.:4.000   Class :character   1st Qu.:541.8
##  Mode  :character   Median :6.000   Mode  :character   Median :583.0
##                     Mean   :5.509                      Mean   :580.8
##                     3rd Qu.:7.000                      3rd Qu.:624.2
##                     Max.   :9.000                      Max.   :903.0
##                     NA's   :3                          NA's   :5
##  iReady_Placement
##  Length:2793
##  Class :character
##  Mode  :character
##
##
##
##
```

The data frame for state test scores has three more rows than the data frame for i-Ready scores. This observation makes sense as the instructions state, "There may be some students with State Test scores who do not have i-Ready scores; however, any student with an i-Ready score has a State Test score." The observation can be confirmed by sub-setting columns and looking at the difference between the data frames.

```
kable(setdiff(select(state_test_scores, colnames(iReady_scores)), iReady_scores))
```

User_ID	School	Student_Grade	iReady_Subject	iReady_Score	iReady_Placement
271370	NA	NA	NA	NA	NA
273016	NA	NA	NA	NA	NA
568363	NA	NA	NA	NA	NA

1. Combine these two datasets based on User_ID (the unique student identifier).

Use `merge()` to combine the two data sets. The three mismatched rows — where there are no i-Ready scores — will be eliminated. Check the `summary()` for the combined data set.

```
data <- merge(iReady_scores, state_test_scores, on="User_ID")
summary(data)
```

```
##     User_ID          School          Student_Grade   iReady_Subject
##  Min.   :100292   Length:2790        Min.   :2.000   Length:2790
##  1st Qu.:315255   Class :character   1st Qu.:4.000   Class :character
##  Median :550539   Mode  :character   Median :6.000   Mode  :character
##  Mean   :549047                      Mean   :5.509
##  3rd Qu.:776441                      3rd Qu.:7.000
##  Max.   :999918                      Max.   :9.000
##
##   iReady_Score   iReady_Placement    Test_Name            Score
##  Min.   :  0.0   Length:2790        Length:2790        Min.   :100.0
##  1st Qu.:541.8   Class :character   Class :character   1st Qu.:366.0
##  Median :583.0   Mode  :character   Mode  :character   Median :417.0
##  Mean   :580.8                                         Mean   :417.7
##  3rd Qu.:624.2                                         3rd Qu.:465.0
##  Max.   :903.0                                         Max.   :999.0
##  NA's   :2                                             NA's   :8
##  Proficiency
##  Length:2790
##  Class :character
##  Mode  :character
##
##
##
##
```

2. Identify any invalid or mismatched data and exclude those students from the analysis. Explain how many students were excluded, and why.

After removing the 3 mismatched observations above, the 2 i-Ready scores and 8 state test scores that were loaded as NA remain in the data set. Filter out these rows so there are valid scores for every student in the analysis. The `summary()` shows that these rows have been eliminated.

```
data <- data %>%
  filter(!is.na(Score) & !is.na(iReady_Score))
summary(data)
```

```
##     User_ID          School          Student_Grade  iReady_Subject
##  Min.   :100292   Length:2780        Min.   :2.00   Length:2780
##  1st Qu.:315475   Class :character   1st Qu.:4.00   Class :character
##  Median :550612   Mode  :character   Median :6.00   Mode  :character
##  Mean   :549336                      Mean   :5.51
##  3rd Qu.:776629                      3rd Qu.:7.00
##  Max.   :999918                      Max.   :9.00
##   iReady_Score   iReady_Placement    Test_Name            Score
##  Min.   :  0.0   Length:2780        Length:2780        Min.   :100.0
##  1st Qu.:541.0   Class :character   Class :character   1st Qu.:366.0
##  Median :583.0   Mode  :character   Mode  :character   Median :417.0
##  Mean   :580.8                                         Mean   :417.7
##  3rd Qu.:624.2                                         3rd Qu.:465.0
##  Max.   :903.0                                         Max.   :999.0
##  Proficiency
##  Length:2780
##  Class :character
##  Mode  :character
##
```

```
##
##
```

Finally, our first point of analysis will be to determine how many students scored at the proficient level on the state test. Check the unique values of the the proficiency column.

```
unique((data$Proficiency))
```

```
## [1] "Proficient"     "Not Proficient" "N/A"
```

There is still some missing data in the data set. Filter out these rows.

```
data <- data %>%
  filter(Proficiency != "N/A")
count(data)
```

```
##      n
## 1 2775
```

The data is now prepared for the analysis. In summary, there are now 2775 students to be included. The following numbers of students were removed from the data:

- 3 students with state scores but no i-Ready scores
- 2 students with a code of P or X instead of an i-Ready score
- 8 students with a code of E or M instead of a state test score
- 5 students with scores on the state test but no proficiency level (proficiency level was "N/A")

Analysis

3. Identify the numbers of students to be included for each grade. Present these counts in a table.

The following table shows the numbers of students to be included for each grade.

```
kable(data %>%
  count("Grade" = Student_Grade), align = "c", caption = "Student Count by Grade")
```

Table 2: Student Count by Grade

Grade	n
2	2
3	459
4	443
5	474
6	514
7	383
8	499
9	1

4. Calculate the percentage of students who are proficient according to the state test for each grade.

Set up a new data frame, grouped by student grade, to calculate proficiency percentages for the i-Ready and state tests. Proficiency for i-Ready is calculated by splitting the iReady_Placement column into columns for grade and placement level within grades. If a student below grade level or on grade at the Early level, that student is not considered proficient. Students who place at Mid- or Late- on grade level, or above grade level, are considered as proficient.

```
percents <- data %>%
  separate(iReady_Placement, c("ir_level", "ir_grade"), sep=" ") %>%
  mutate(st_prof = ifelse(data$Proficiency == "Proficient",1,0)) %>%
  mutate(ir_prof = case_when(
    ir_grade > Student_Grade ~ 1,
    ir_grade == Student_Grade & ir_level != "Early" ~ 1,
    TRUE ~ 0
  )) %>%
  group_by(Student_Grade) %>%
  summarise(Count = n(),
            State_Test_Percent_Proficient = (sum(st_prof) / n() * 100),
            iReady_Percent_Proficient = (sum(ir_prof) /n() * 100))
```

Display percentages of proficient students for the state test.

```
st_percents <- percents%>%
  group_by("Grade" = Student_Grade) %>%
  summarise("Percent Proficient (%)" = State_Test_Percent_Proficient)
kable(st_percents, align = "c", caption="Percent Proficent, State Test")
```

Table 3: Percent Proficent, State Test

Grade	Percent Proficient (%)
2	50.00000
3	52.94118
4	42.88939
5	41.35021
6	41.05058
7	23.23760
8	20.24048
9	100.00000

5. Calculate the percentage of students who are proficient according to i-Ready Diagnostic for each grade. (Refer to the bullet points above for details on proficiency in i-Ready)

```
ir_percents <- percents%>%
  group_by("Grade" = Student_Grade) %>%
  summarise("Percent Proficient (%)" = iReady_Percent_Proficient)
kable(ir_percents, align="c", caption="Percent Proficent, i-Ready")
```

Table 4: Percent Proficent, i-Ready

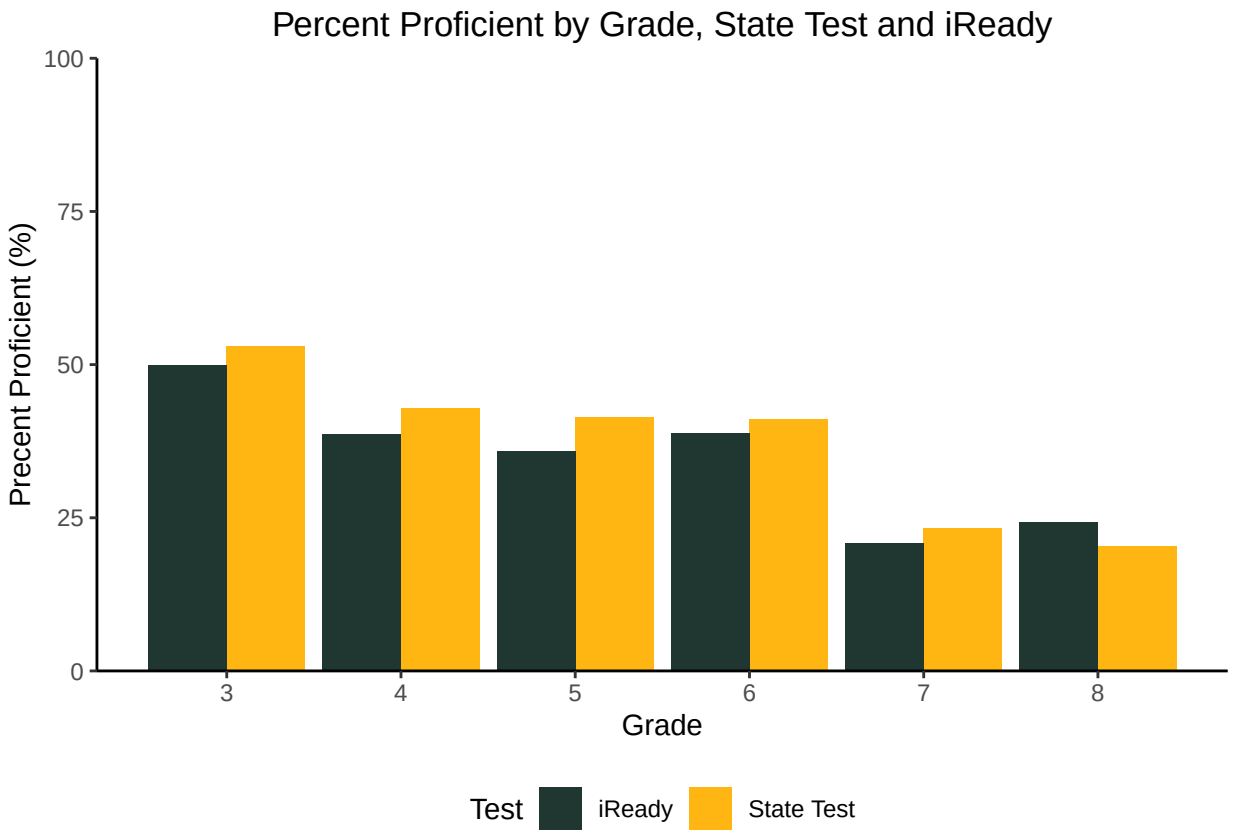
Grade	Percent Proficient (%)
2	50.00000
3	49.89107
4	38.60045
5	35.86498
6	38.71595
7	20.88773
8	24.24850
9	0.00000

6. Create one visual representation. Your visual should show the percentage of students who are proficient according to each assessment, for each grade.

Pivot the data to a long format to facilitate grouping and create the visualization.

```
percents_longer <-percents %>%
  pivot_longer(cols = c(State_Test_Percent_Proficient, iReady_Percent_Proficient), names_to="Test", valu

ggplot(filter(percents_longer, Student_Grade!=2 & Student_Grade!=9),
       aes(x=Student_Grade, y=Percent_Proficient, fill=Test)) +
  geom_bar(stat="identity", position = "dodge")  +
  theme_classic() +
         scale_x_continuous(breaks=c(3:8)) +
  scale_y_continuous(expand = c(0, 0), limits=c(0,100)) +
  theme(legend.position="bottom", plot.title = element_text(hjust = 0.5)) +
  ggtitle("Percent Proficient by Grade, State Test and iReady") +
  xlab("Grade") + ylab("Precent Proficient (%)") +
  scale_fill_manual(name = "Test", labels = c("iReady","State Test"), values=c("#203731", "#FFB612"))
```



Note that grades 2 and 9 are not included in the visualization. The data set contained 2 students at grade 2 and 1 student at grade 9. Student counts at grades 2 and 9 are not large enough to make any general claims about the tests and may also lead to misleading visual representation of the percentages.

7. Write a short paragraph explaining why you chose the visualization you used, as well as the interpretations you would make from the data.

I chose a grouped bar chart as the visualization for this data because it is relatively simple to produce, is relatively simple to interpret, and presents a substantial amount of information from the data. The grouped bar chart allows for easy comparison of percent proficient within grades as well as comparison across grade

levels. In general, proficiency percentages for each grade for each test range from about 20% to about 50%. For every grade except grade 8, the percentage of proficient students was slightly higher for the state test than that for i-Ready. Also, one can observe a slight downward trend in the percentages of proficient students as the grade level increases for both tests. Ultimately, the percentage of proficient students on each test is near to that of the other test for each grade. Proficiency percentages are only one way of judging correlations between the two tests; further investigation can be done using correlations and regressions between scale scores.

8. Submit your code with your response.

The code for this analysis should be visible in this notebook, If not, or to hide the code and read through the analysis, click Code in the upper right hand corner or click the Code button next to each block.