MA578 Bayesian Statistics Final Project

Probability and Statistics course Final Exam score analysis.

Bingtian Ye

2023-12-9

Introduction:

Each student will have a different performance in the Probability and statistics course, which is specifically reflected in their grade. In this final project, I want to explore which of the usual exams (The dataset has three usual exams), attendance, and homework indicators have the greatest impact on students' final exams. Since this course is so relevant to us, I thought it would be interesting to study this topic. At the same time, I think all students care about their scores. In addition, studying this issue has certain pedagogical implications. For example, eventually, we can build a Bayesian regression model or other models. To determine which indicators, have a greater impact on students' final grades. This way, students can draw attention when they find themselves with low values on these indicators.

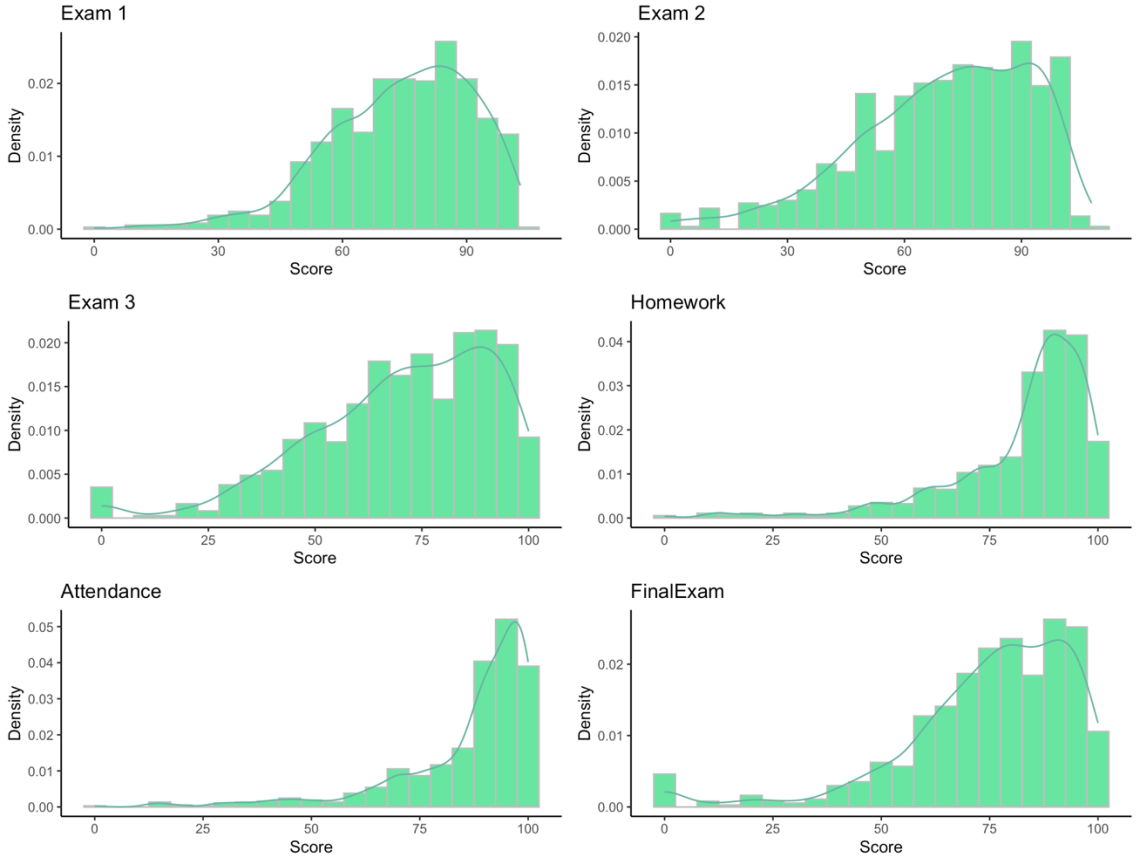
Dataset:

The dataset is the Probability and Statistics course from Kaggle (https://www.kaggle.com/datasets/indikawickramasinghe/probability-and-statistics-course-performance), which includes students' performance in a college-level course in Probability and Statistics from 2016 to 2022. Also, the dataset has nine features, which are Year, Semester, Exam.1, Exam.2, Exam.3, Homework, Attendance, FinalExam, FinalGrade.

EDA:

1. Distribution of variable

First, I want to draw the distribution of each variable, so that I can choose the prior distribution and regression family and link.



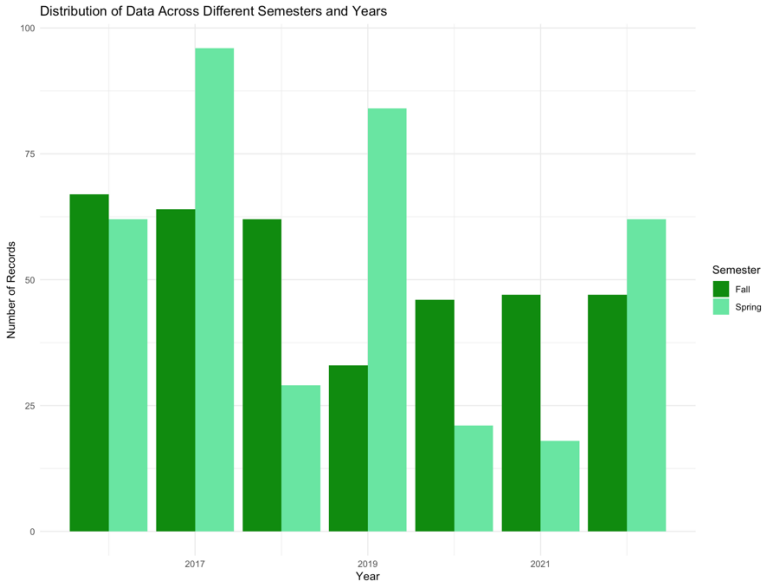
Although the data has a certain left skew, it still approximately obeys a normal distribution. So, I can choose a Multivariate Gaussian distribution as a prior distribution and use Bayesian Linear Regression to build the model.

2. Choose variable

Plot correlation coefficients Matrix to determine which variables should be selected.

It looks like all predictors have some degree of correlation with the predicted variable, which means I can select all predictors. At the same time, we can see that the correlation coefficient of Exam.3 is greater than that of Exam.1 and Exam.2. This is also more in line with our common sense because Exam.3 is closest to FinalExam.

3. Determine whether historical data can be used as a priori.



The distribution of the data shows the amount of data across different academic years and semesters. We can see that the number of records varies for each semester, but there is a certain amount of data for each semester. This means that we can use data from the previous semester or semesters as prior information.

Modeling

Use the mean and variance of the previous year's data as the prior distribution. Establish a weight ρ:

$$\rho = f(t, \kappa_0),$$
$$\mu_0 = \rho_1 \mu_0^1 + \rho_2 \mu_0^2 + \cdots,$$

$$\sigma_0^2 = \rho_1 \sigma_0^{2\{1\}} + \rho_2 \sigma_0^{2\{2\}} + \cdots.$$

Among them, t is the number of semester differences, and κ_j is the sample size of different prior samples. For example, if we want to get 2017 Spring's prior distribution by using 2016 Spring and 2016 Fall. The parameters of prior distribution are as follows:

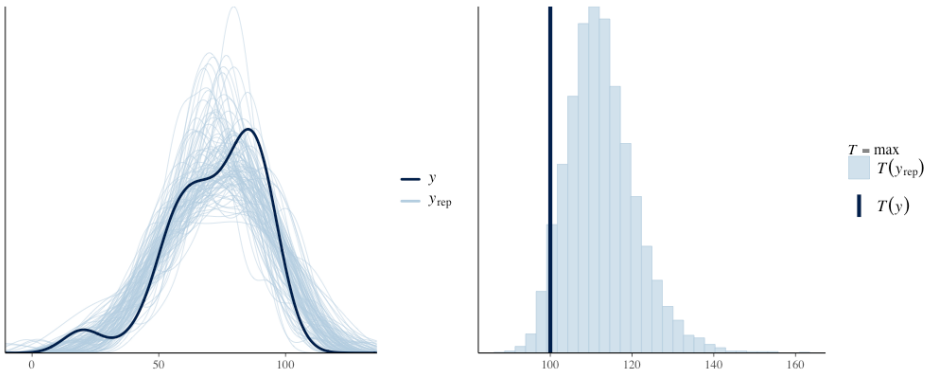
$$\mu_0 = \rho_1 \mu_0^1 + \rho_2 \mu_0^2,$$
$$\sigma_0^2 = \rho_1 \sigma_0^{2\{1\}} + \rho_2 \sigma_0^{2\{2\}}.$$
$$\rho_1 = \left(\alpha + \beta \, {\kappa_1}/{\kappa_1 + \kappa_2}\right)^{t=2},$$
$$\rho_2 = \left(\alpha + \beta \, {\kappa_2}/{\kappa_1 + \kappa_2}\right)^{t=1}.$$

α is the parameter of the number of semester differences, which is between 0 and 1, β is the parameter of sample size. ρ_1 and ρ_2 is less than 1. Let $\alpha = 0.5, \beta = 0.1$ in this model (In practice, cross-validation can be used). Due to page limit, only 2022Fall is modeled as a sample variable here, and the loss function is MSE. Create three models respectively:

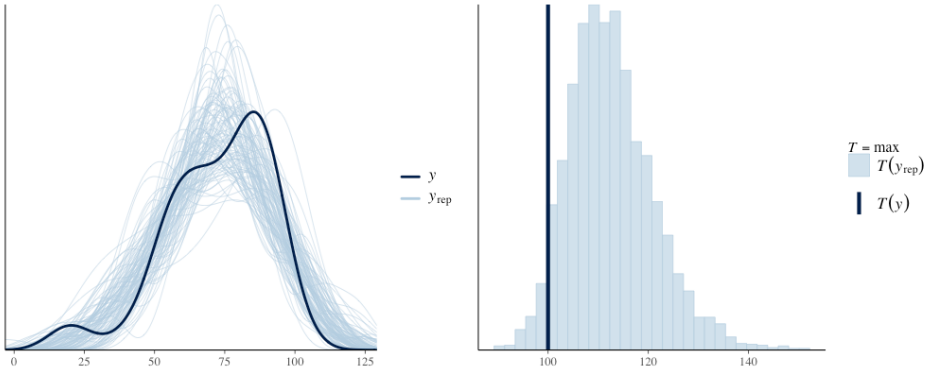
1. Only use 2022Spring as prior.
2. All previous exams are used as prior.
3. Establish three models by using only the previous Fall data as the prior.

Model1: Only use 2022Spring as prior

```
Family: gaussian
  Links: mu = identity; sigma = identity
Formula: FinalExam ~ Exam.1 + Exam.2 + Exam.3 + Homework + Attendance
   Data: data[data$Year == 2022 & data$Semester == "Fall",  (Number of observations: 47)
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000
Population-Level Effects:
           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept     19.32     22.49   -25.67    63.37 1.00     4326     2723
Exam.1         0.25      0.27    -0.29     0.81 1.00     3902     2581
Exam.2         0.45      0.16     0.13     0.76 1.00     3986     2795
Exam.3         0.19      0.20    -0.21     0.60 1.00     4129     2701
Homework      -0.01      0.15    -0.30     0.28 1.00     4303     2323
Attendance    -0.16      0.21    -0.57     0.25 1.01     4198     2268
Family Specific Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sigma    13.13      1.50    10.51    16.47 1.00 3820     2828
```



It can be seen that all Rhat of the model are close to 1, indicating that the parameters and model are convergent, and the value of ESS is large, indicating that the model works well. However, judging from the ppc chart, the results of the model still have some problems of overestimation of the maximum value. Add cross terms to get a new model, see Appendix 1. Also draw ppc plot as follow.



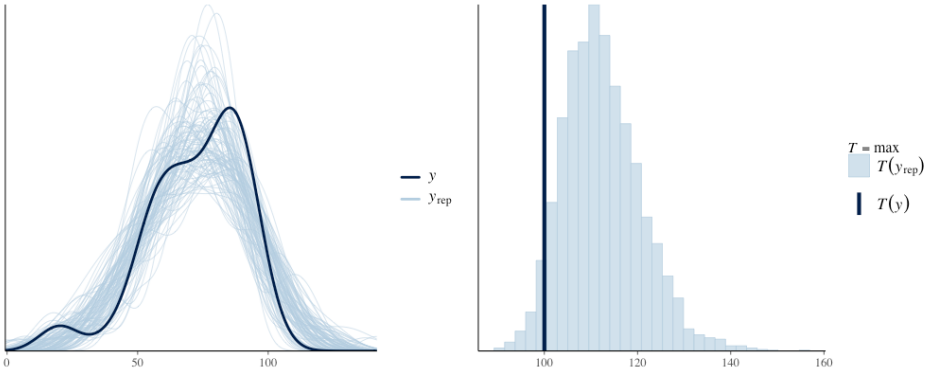
It is difficult to compare which of the two models is better based on the results of the ppc chart and the Rstudio session aborted when further calculating the Bayes factor. Therefore, LOO is used to compare the two models, and the results are as follows:

```
elpd_diff se_diff
model1  0.0       0.0
model2 -4.8       2.1
```

This shows that the model without interaction terms is better than the model with interaction terms. So, I will not use interaction terms in subsequent models.

Model2: All previous exams are used as prior

```
Population-Level Effects:
           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept     19.10     22.64   -26.32    63.57 1.00     4590     3357
Exam.1         0.24      0.27    -0.29     0.77 1.00     3561     2728
Exam.2         0.45      0.17     0.13     0.78 1.00     4069     2656
Exam.3         0.20      0.20    -0.18     0.58 1.00     3680     2647
Homework      -0.02      0.15    -0.31     0.26 1.00     4479     2900
Attendance    -0.16      0.21    -0.58     0.27 1.00     4486     2427
Family Specific Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sigma    13.14      1.48    10.64    16.35 1.00     3581     2671
```

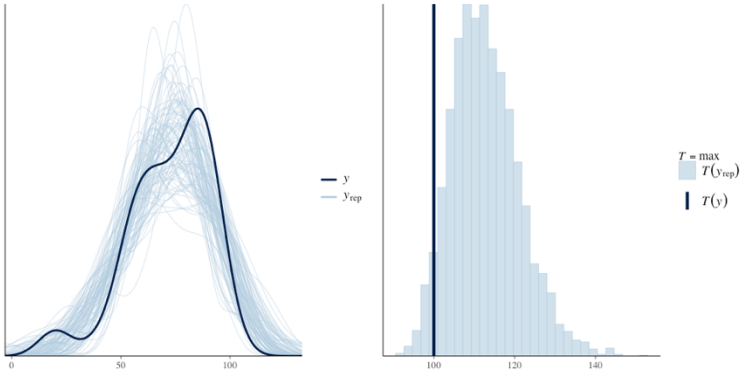


Model3:Establish three models by using only the previous Fall data as the prior.

```
Population-Level Effects:
           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept     18.64     22.49   -24.66    63.33 1.00     4898     3569
Exam.1         0.23      0.27    -0.30     0.76 1.00     3785     2448
Exam.2         0.45      0.16     0.13     0.77 1.00     4458     2625
Exam.3         0.20      0.20    -0.17     0.59 1.00     3996     2826
Homework      -0.01      0.15    -0.31     0.28 1.00     4620     2937
Attendance    -0.15      0.21    -0.58     0.27 1.00     5127     2846

Family Specific Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
```

```
sigma    13.11      1.48    10.58    16.29 1.00     3252     2646
```



Model Comparison

By comparing several models using LOO, the results are as follows:

```
                   elpd_diff se_diff
model2              0.0       0.0
model3             -0.1       0.3
model1             -0.3       0.2
model1_interaction -5.1       2.0
```

According to the results, model2 is better than model3, then model1 and model1 with the interaction term added.

It was found that the model using all previous period data as the prior had the best results, followed by the model using only previous period data in the same season as the prior, and the third model using only the previous period data and using interaction terms had the worst results. (I used interaction terms for all three models, and the results were not satisfactory).

Conclusion

According to the results, it can be concluded that the impact of season on test scores is not significant, and there are certain differences in scores from semester to semester, which makes using all previous data as a priori better. All models indicate that the coefficients on all three test scores are positive, while the coefficients on homework and attendance are negative. This may mean that students with good final exam scores are often those students who have a good exam status and mentality (that is, students who perform better in exams), rather than those students who have high attendance rates and better homework assignments.

The negative parameter for attendance can be understood to mean that these students have good self-learning ability and have mastered relevant knowledge after class. This self-study ability may enable them to review better before the Final Exam, thereby leading to higher grades. Of course, this behavior is not worth promoting. However, it is worth noting that the intervals of all parameters of the four models include 0, which may indicate that all parameters are not significant. I tried to adjust the number of iterations and other parameters of the model, but the results were basically not significant. Therefore, we can also think that it is difficult to judge a student's final exam score from his previous performance. In other words, even if a student has not performed satisfactorily in previous exams, attendance, and homework, it does not mean that his final grade will definitely be poor.

Appendix

1.Model summary of Only use 2022Spring as prior and add interaction

```
Family: gaussian
  Links: mu = identity; sigma = identity
Formula: FinalExam ~ Exam.1 * Exam.2 * Exam.3 + Homework * Attendance
   Data: data[data$Year == 2022 & data$Semester == "Fall",  (Number of observations: 47)
  Draws: 4 chains, each with iter = 10000; warmup = 5000; thin = 1;
         total post-warmup draws = 20000

Population-Level Effects:
                     Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept             -106.22    206.63  -534.01   274.52 1.00     5097     7682
Exam.1                   2.75      2.27    -1.52     7.43 1.00     5428     8190
Exam.2                   3.75      2.40    -0.69     8.66 1.00     5293     7804
Exam.3                   3.63      2.76    -1.56     9.28 1.00     5328     8675
Homework                -0.84      1.42    -3.57     2.02 1.00     6726     8438
Attendance              -1.00      1.53    -3.91     2.07 1.00     6779     8192
Exam.1:Exam.2           -0.04      0.03    -0.11     0.02 1.00     5584     8013
Exam.1:Exam.3           -0.04      0.04    -0.12     0.03 1.00     5266     8704
Exam.2:Exam.3           -0.05      0.04    -0.13     0.01 1.00     5139     8509
Homework:Attendance      0.01      0.02    -0.02     0.04 1.00     6762     8499
Exam.1:Exam.2:Exam.3     0.00      0.00    -0.00     0.00 1.00     4704     7167

Family Specific Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sigma    13.80      1.76    10.89    17.82 1.00     8417    11245

Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
and Tail_ESS are effective sample size measures, and Rhat is the potential
scale reduction factor on split chains (at convergence, Rhat = 1).
```

2.Code

```
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
pacman::p_load("bayesplot","knitr","ggplot2","rstanarm","brms","ggpubr","dplyr","gridExtra","corrplot")
```


```{r}
data <- read.csv("ProbStat.csv")
```
EDA
```{r}
#distribution of variable
p1 <- ggplot(data, aes(x = Exam.1)) +
  geom_histogram(aes(y = ..density..), binwidth = 5, fill = "#69e5a2", color = "grey") +
  geom_density(alpha = .2, color = "#69b3a2") +
  labs(title = "Exam 1", x = "Score", y = "Density") +
   theme_classic()

p2 <- ggplot(data, aes(x = Exam.2)) +
  geom_histogram(aes(y = ..density..), binwidth = 5, fill = "#69e5a2", color = "grey") +
  geom_density(alpha = .2, color = "#69b3a2") +
  labs(title = "Exam 2", x = "Score", y = "Density") +
   theme_classic()
```
```

```
p3 <- ggplot(data, aes(x = Exam.3)) +
  geom_histogram(aes(y = ..density..), binwidth = 5, fill = "#69e5a2", color = "grey") +
  geom_density(alpha = .2, color = "#69b3a2") +
  labs(title = "Exam 3", x = "Score", y = "Density") +
   theme_classic()

p4 <- ggplot(data, aes(x = Homework)) +
  geom_histogram(aes(y = ..density..), binwidth = 5, fill = "#69e5a2", color = "grey") +
  geom_density(alpha = .2, color = "#69b3a2") +
  labs(title = "Homework", x = "Score", y = "Density") +
   theme_classic()

p5 <- ggplot(data, aes(x = Attendance)) +
  geom_histogram(aes(y = ..density..), binwidth = 5, fill = "#69e5a2", color = "grey") +
  geom_density(alpha = .2, color = "#69b3a2") +
  labs(title = "Attendance", x = "Score", y = "Density") +
   theme_classic()

p6 <- ggplot(data, aes(x = FinalExam)) +
  geom_histogram(aes(y = ..density..), binwidth = 5, fill = "#69e5a2", color = "grey") +
  geom_density(alpha = .2, color = "#69b3a2") +
  labs(title = "FinalExam", x = "Score", y = "Density") +
   theme_classic()

grid.arrange(p1, p2, p3, p4, p5, p6, ncol=2)

cor_matrix <- cor(data[, c("Exam.1", "Exam.2", "Exam.3", "Homework", "Attendance", "FinalExam")])

corrplot(cor_matrix, method = "color", col = colorRampPalette(c("#6BAED6", "#FFFFFF", "#FD8D3C"))(200),
         type = "upper", addCoef.col = "black", tl.col = "black", tl.srt = 45,
         diag = TRUE)

ggplot(data, aes(x = Year, fill = Semester)) +
  geom_bar(position = "dodge") +
  scale_fill_manual(values = c("Fall" = "#0F8B0F", "Spring" = "#69e5a2")) +
  labs(title = "Distribution of Data Across Different Semesters and Years",
       x = "Year", y = "Number of Records") +
  theme_minimal()
```
#tidy data
```{r}
pivot_data <- data %>%
  group_by(Year, Semester) %>%
  summarise(
    Exam1_Mean = mean(Exam.1, na.rm = TRUE),
    Exam1_Var = var(Exam.1, na.rm = TRUE),
    Exam2_Mean = mean(Exam.2, na.rm = TRUE),
    Exam2_Var = var(Exam.2, na.rm = TRUE),
```

```
    Exam3_Mean = mean(Exam.3, na.rm = TRUE),
    Exam3_Var = var(Exam.3, na.rm = TRUE),
    Homework_Mean = mean(Homework, na.rm = TRUE),
    Homework_Var = var(Homework, na.rm = TRUE),
    Attendance_Mean = mean(Attendance, na.rm = TRUE),
    Attendance_Var = var(Attendance, na.rm = TRUE),
    FinalExam_Mean = mean(FinalExam, na.rm = TRUE),
    FinalExam_Var = var(FinalExam, na.rm = TRUE),
    N = n()
  ) %>%
  ungroup()
```
##modeling
###model1
```{r}
prior <- c(
  prior(normal(72.9,sqrt(407.8)), class = "Intercept"),
  prior(normal(74.5,sqrt(227.5)), class = "b", coef = "Exam.1"),
  prior(normal(70.0,sqrt(537.0)), class = "b", coef = "Exam.2"),
  prior(normal(61.7, sqrt(629.5)), class = "b", coef = "Exam.3"),
  prior(normal(82.0, sqrt(300.5)), class = "b", coef = "Homework"),
  prior(normal(73.5, sqrt(496.0)), class = "b", coef = "Attendance")
)
model1 <- brm(
  FinalExam ~ Exam.1 + Exam.2 + Exam.3 + Homework + Attendance,
  data = data[data$Year == 2022 & data$Semester == "Fall", ],
  prior = prior,
  family = gaussian(),
  chains = 4,
  iter = 2000,
  warmup = 1000
)
summary(model1)
yrep <- posterior_predict(model1)
mp1 <- ppc_dens_overlay(data[data$Year == 2022 & data$Semester == "Fall","FinalExam"], yrep[1:100,])
mp2 <- ppc_stat(data[data$Year == 2022 & data$Semester == "Fall","FinalExam"], yrep, stat="max")
grid.arrange(mp1, mp2, ncol=2)
```
###model1 interaction
```{r}
prior <- c(
  prior(normal(72.9, sqrt(407.8)), class = "Intercept"),
  prior(normal(74.5, sqrt(227.5)), class = "b", coef = "Exam.1"),
  prior(normal(70.0, sqrt(537.0)), class = "b", coef = "Exam.2"),
  prior(normal(61.7, sqrt(629.5)), class = "b", coef = "Exam.3"),
  prior(normal(82.0, sqrt(300.5)), class = "b", coef = "Homework"),
  prior(normal(73.5, sqrt(496.0)), class = "b", coef = "Attendance"),
  prior(normal(0, 10), class = "b", coef = "Exam.1:Exam.2"),
  prior(normal(0, 10), class = "b", coef = "Exam.1:Exam.3"),
```

```
  prior(normal(0, 10), class = "b", coef = "Exam.2:Exam.3"),
  prior(normal(0, 10), class = "b", coef = "Exam.1:Exam.2:Exam.3"),
  prior(normal(0, 10), class = "b", coef = "Homework:Attendance")
)
model1_interaction <- brm(
  FinalExam ~ Exam.1 * Exam.2 * Exam.3 + Homework * Attendance,
  data = data[data$Year == 2022 & data$Semester == "Fall", ],
  prior = prior,
  family = gaussian(),
  chains = 4,
  iter = 10000,
  warmup = 5000
)
summary(model1_interaction)
yrep <- posterior_predict(model1_interaction)
mp1 <- ppc_dens_overlay(data[data$Year == 2022 & data$Semester == "Fall","FinalExam"], yrep[1:100,])
mp2 <- ppc_stat(data[data$Year == 2022 & data$Semester == "Fall","FinalExam"], yrep, stat="max")
grid.arrange(mp1, mp2, ncol=2)
```
```{r}
loo1 <- loo(model1)
loo2 <- loo(model2)
loo_compare(loo1, loo2)
```