Final Exam

Dan Cher

2024-12-17; 12:30 PM ~5:20 PM (with breaks)

Mechanical Turk Analysis

1. First let’s use BMI as our outcome variable of interest. We are interested in understanding if master workers are different than other workers when it comes to BMI. Fit a standard (classical) linear regression model to this data. You can go ahead and just control for all of the covariates (except exercise).

```r
# fit model
model <- lm(bmi ~ TurkMaster + agecat + race_cat + female + income_cat, data = mechturk_data)

# view summary stats
summary(model)
```

a. Interpret the results. We are particularly interested in the results for the TurkMaster. Be sure to provide a correct interpretation for the p-values/stars. What conclusions do you draw about whether master workers are different when it comes to BMI?

```
##
## Call:
## lm(formula = bmi ~ TurkMaster + agecat + race_cat + female +
##     income_cat, data = mechturk_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.968  -4.368  -1.559   2.454  38.050
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  27.3916     0.7188  38.106   <2e-16 ***
## TurkMaster    1.0635     0.4391   2.422   0.0156 *
## agecat        0.2722     0.2176   1.251   0.2112
## race_cat     -0.5488     0.2134  -2.571   0.0103 *
## female       -0.3298     0.4364  -0.756   0.4499
## income_cat   -0.3177     0.1751  -1.815   0.0699 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 6.708 on 978 degrees of freedom
## Multiple R-squared:  0.02035,    Adjusted R-squared:  0.01534
## F-statistic: 4.062 on 5 and 978 DF,  p-value: 0.001179
```

Overall, the adjusted R-squared is quite low (~0.02). Only ~2% of the variance in BMI is explained using the linear model defined above. This suggests that are other factors that are impactful in determining BMI not integrated in the model.

Understanding this limitation, the results of specific variables show a few variables of interest that are statistically significant. Specifically TurkMaster and race_cat have p-values below 0.05. P-value meaning that the coefficient results for these variables that we empirically derived have a probability lower than 5%, assuming the null hypothesis, which is that these regression coefficients = 0, is true. Since we specifically care about TurkMaster, we can say that TurkMaster has a strong positive relationship in association with BMI of Mechanical Turk workers. Whether the worker is a master is associated with a 1.0635 unit increase in BMI compared to non-masters.

b. Thinking about this data, discuss two ways that the inferences you draw from this standard regression might be invalid. Focus on the kinds of assumptions we need to make to do inference in a classical/frequentist setting.

1. Autocorrelation issues The frequentist model assumes that residuals are i.i.d (independent and identically distributed). In this case the errors are likely correlated based on the region of living, or demographics of the workers that are not captured in the data. Once this assumption is violated, the inferences drawn from this model are no longer valid. To deal with this, you would likely need to incorporate clustered standard errors.
2. Multicollinearity The frequentist model assumes that variables are not correlated. Age, income and race are all variables that are known to be highly correlated. Highly correlated variables make the results of the model not reliable. Incorporating a bayesian framework with prior distributions can help deal with this type of issue.

More generally, the inferences drawn from this type of framework are hand-cuffed in their direct interpretability. Bayesian frameworks allow us to more directly express the likelihood of our results, rather than relying on a long-run frequency that requires an unbiased sampling process (even in ways that may not be relevant to our data).

2. In problem 1, you made use of the Null Hypothesis Significance Testing framework. Answer the questions below focusing again on the TurkMaster variable.

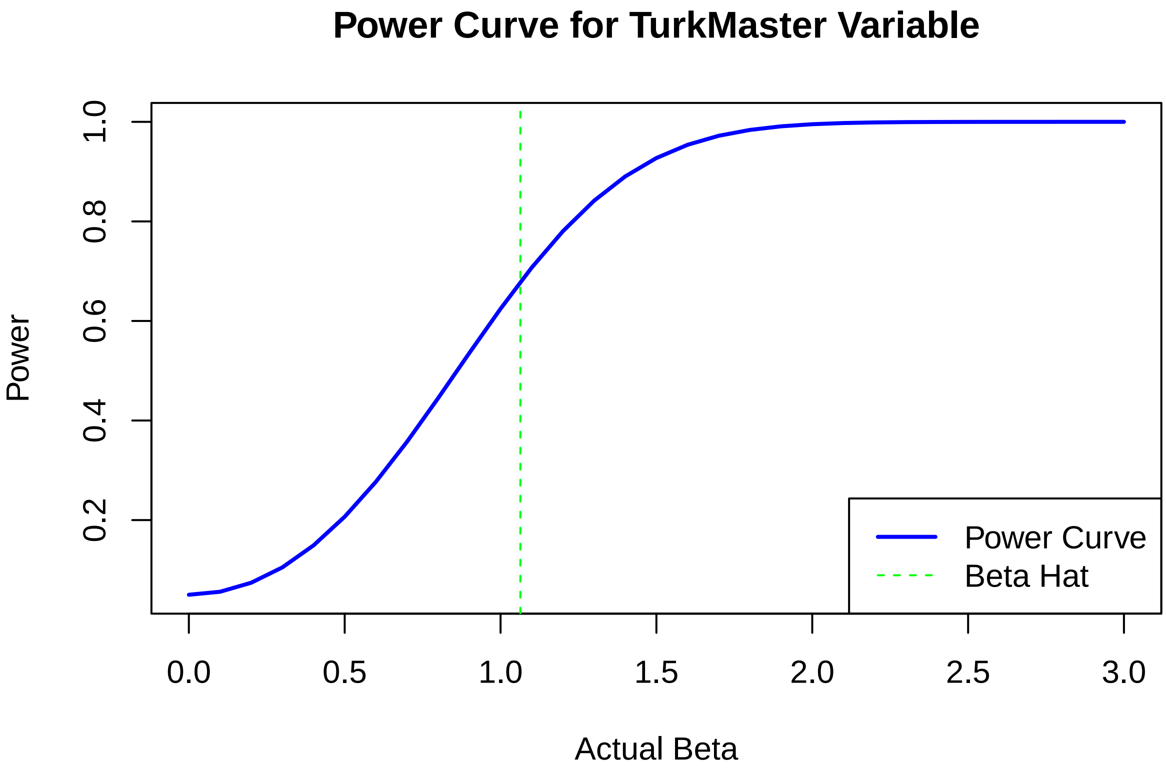
a. Define the null and alternative hypothesis for this test. The null hypothesis is that the coefficient of the Turkmaster variable == 0. This means there is no difference between masters and non-masters. The alternative hypothesis is that there is a difference between masters and non-masters.

b. Explain what the p-value is and what it means in the context of this regression. In this context, p-value means that the coefficient results for Turkmaster that we empirically derived have a probability lower than 5% of being this extreme or more, assuming the null hypothesis, which is that Turkmaster regression coefficient = 0, is true.

c. What is the type-I error rate of the test you used to determine whether or not the variable was "significant." The type-I error rate is probability of rejecting the null hypothesis when it is actually true. In this context that means the likelihood of concluding there is a signficant association between Turkmaster and BMI when no association actually exists. Here we used 0.05 (5%).

```r
# params
beta_hat <- 1.0635
SE <- 0.4391
alpha <- 0.05
t_critical <- qnorm(1 - alpha / 2)

# power for test-statistic found in glm
power_test <- 1 - (pnorm(t_critical, mean = beta_hat / SE, sd = 1) -
              pnorm(-t_critical, mean = beta_hat / SE, sd = 1))

# power curve for different betas
beta_range <- seq(0, 3, by = 0.1)
power <- 1 - (pnorm(t_critical, mean = beta_range / SE, sd = 1) -
              pnorm(-t_critical, mean = beta_range / SE, sd = 1))

# plot
plot(beta_range, power, type = "l", col = "blue", lwd = 2,
     xlab = "Actual Beta", ylab = "Power",
     main = "Power Curve for TurkMaster Variable")
abline(v = beta_hat, col = "green", lty = 2)
legend("bottomright", legend = c("Power Curve", "Beta Hat"),
       col = c("blue",  "green"), lty = c(1, 2), lwd = c(2, 1))
```

d. The significance test here is calculated simply by taking the coefficient and dividing by the standard error to derive a t-statistic with n-p degrees of freedom. Assume that the standard errors are fixed and plot the "power curve" for different values of β. Is your test well powered?



Why or why not?

```r
cat("Power:", power_test)
```

```
## Power: 0.677978
```

The power of our test is 0.678. This is generally considered not well powered, as 0.80 is the threshold for a well-powered test. There is a 67% chance of correctly rejecting the null hypothesis if the true effect size of the coefficient = 1.0635.

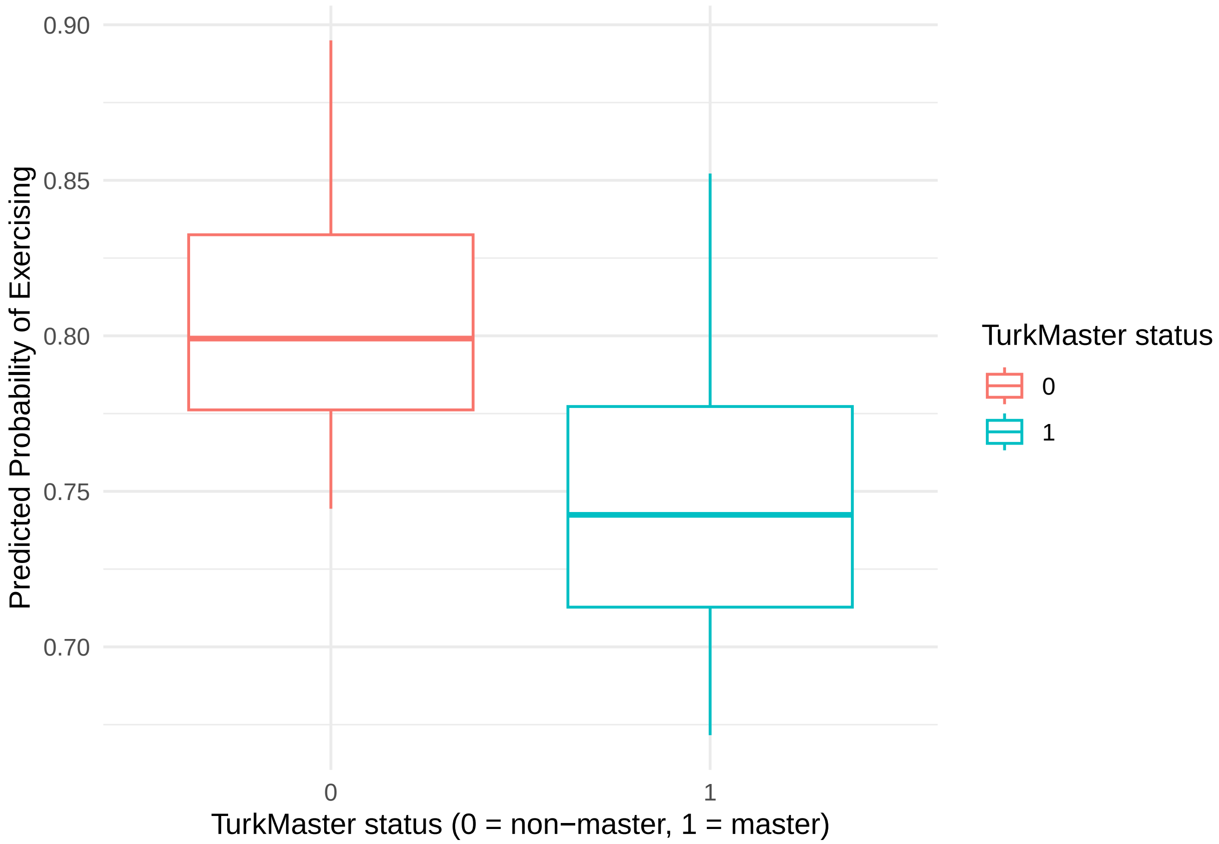
3. Your co-author theorizes that the differences in BMI could be a result of differing lifestyles. Perhaps master workers are more (or less) likely to exercise.

```r
# remove NAs
mechturk_data_clean <- mechturk_data[!is.na(mechturk_data$exercise), ]

# logistic regression b/c exercise is binary variable
glm_model <- glm(exercise ~ TurkMaster + agecat + race_cat + female + income_cat,
                 data = mechturk_data_clean,
                 family = binomial(link = "logit"))

summary(glm_model)
```

a. Fit an appropriate generalized linear model using exercise as the dependent variable.

```
##
## Call:
## glm(formula = exercise ~ TurkMaster + agecat + race_cat + female +
##     income_cat, family = binomial(link = "logit"), data = mechturk_data_clean)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.93803    0.26179   3.583 0.000339 ***
## TurkMaster  -0.35353    0.15690  -2.253 0.024244 *
## agecat      -0.01877    0.07847  -0.239 0.810995
## race_cat     0.04042    0.08073   0.501 0.616574
## female       0.19339    0.15935   1.214 0.224894
## income_cat   0.16562    0.06591   2.513 0.011973 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1033.3  on 973  degrees of freedom
## Residual deviance: 1019.6  on 968  degrees of freedom
## AIC: 1031.6
##
## Number of Fisher Scoring iterations: 4
```

```r
mechturk_data_clean$predicted_probs <- predict(glm_model, type = "response")

# plot
ggplot(mechturk_data_clean, aes(x = factor(TurkMaster), y = predicted_probs, color = factor(TurkMaster)
  geom_boxplot() +
  labs(x = "TurkMaster status (0 = non-master, 1 = master)",
       y = "Predicted Probability of Exercising",
       color = "TurkMaster status") +
  theme_minimal()
```



```
cat('Odds ratio of TurkMaster:', exp(glm_model$coefficients[2]))
```

```
## Odds ratio of TurkMaster: 0.7022023
```

b. Write one paragraph interpreting these results to the readers of a hypothetical article. Explain what the "effect size" is. Provide a plot to help the reader understand what your model says. To examine the relationship between various factors and their effects on exercise for employees of mechanical turk, we conducted an analysis using a logistic regression model. The model reveals the likelihood of exercising is significantly associated with whether the employee is a "master" worker as compared to a regular worker. Specifically, masters are less likely to exercise with an odds ratio of 0.70, meaning they are 30% less likely to exercise compared to their non-master counterparts. Our plot shows the predicted probabilities of exercising based on whether the worker is master/non-master. To interpret the model we typically care about the "effect size" which for a logistic regression refers to the odds ratio for each predictor. This quantifies the change in the odds of exercising associated with a one unit-change in our variable – for our case, non-master as compared to master. The attached boxplot shows the differing distributions of predicted probabilities of exercising on whether the worker is master or not.

United Nations Security Council

4. Using the lm_robust function, fit a model that (a) includes fixed effects for each country and separate fixed effects for each year and (b) has clustered standard errors at the country level. The only other covariate you need is l_pctagreeus_unga.

```r
# model w/ fixed effects on country & year + clustered SE
fe_model <- lm_robust(DependentVariable ~ l_pctagreeus_unga + factor(wdicode) + factor(year),
                   data = unsc_data,
                   clusters = unsc_data$wdicode)

summary(fe_model)
```

```
##
## Call:
## lm_robust(formula = DependentVariable ~ l_pctagreeus_unga + factor(wdicode) +
##     factor(year), data = unsc_data, clusters = unsc_data$wdicode)
##
## Standard error type:  CR2
##
## Coefficients:
##                      Estimate Std. Error  t value  Pr(>|t|) CI Lower   CI Upper
## (Intercept)          0.386759    0.09848  3.92726 5.513e-03  0.15466  6.189e-01
## l_pctagreeus_unga   -0.246135    0.14977 -1.64337 1.295e-01 -0.57712  8.485e-02
## factor(wdicode)AUS -0.014083    0.04813 -0.29259 7.748e-01 -0.11877  9.060e-02
## factor(wdicode)BEL -0.117756    0.04529 -2.59994 2.221e-02 -0.21574 -1.977e-02
## factor(wdicode)BOL -0.053178    0.03298 -1.61255 2.071e-01 -0.15937  5.302e-02
## factor(wdicode)BRA -0.074161    0.03261 -2.27443 4.184e-02 -0.14510 -3.220e-03
## factor(wdicode)CAN -0.034454    0.02318 -1.48634 1.687e-01 -0.08625  1.735e-02
## factor(wdicode)CHL -0.066677    0.05348 -1.24669 2.419e-01 -0.18644  5.309e-02
## factor(wdicode)CIV -0.017736    0.03357 -0.52838 6.349e-01 -0.12651  9.104e-02
## factor(wdicode)COL -0.089451    0.02960 -3.02164 9.623e-03 -0.15327 -2.563e-02
## factor(wdicode)CSK  0.290956    0.07179  4.05302 1.502e-02  0.09271  4.892e-01
## factor(wdicode)CUB -0.076935    0.02900 -2.65282 2.061e-02 -0.13991 -1.396e-02
## factor(wdicode)DNK -0.024955    0.04372 -0.57086 5.865e-01 -0.12903  7.912e-02
## factor(wdicode)ECU -0.115495    0.04211 -2.74279 1.880e-02 -0.20794 -2.305e-02
## factor(wdicode)EGY  0.046509    0.07774  0.59823 5.620e-01 -0.12487  2.179e-01
## factor(wdicode)GHA -0.021861    0.08702 -0.25122 8.082e-01 -0.22387  1.802e-01
## factor(wdicode)GRC -0.110383    0.04818 -2.29091 5.007e-02 -0.22081  4.425e-05
## factor(wdicode)IND  0.238168    0.06665  3.57337 4.184e-03  0.09197  3.844e-01
## factor(wdicode)IRL -0.037271    0.05815 -0.64090 5.454e-01 -0.17969  1.051e-01
## factor(wdicode)IRN -0.110316    0.07346 -1.50180 1.659e-01 -0.27536  5.472e-02
## factor(wdicode)IRQ -0.019300    0.03975 -0.48551 6.387e-01 -0.10897  7.037e-02
## factor(wdicode)ITA -0.076224    0.02209 -3.45119 1.643e-02 -0.13195 -2.050e-02
## factor(wdicode)JPN -0.021842    0.01994 -1.09549 3.176e-01 -0.07130  2.762e-02
## factor(wdicode)LBN  0.235692    0.06085  3.87346 5.893e-03  0.09233  3.790e-01
## factor(wdicode)LBR  0.054777    0.09297  0.58921 5.776e-01 -0.17385  2.834e-01
## factor(wdicode)LKA  0.107538    0.07880  1.36473 2.076e-01 -0.07256  2.876e-01
## factor(wdicode)MAR  0.103243    0.06797  1.51885 1.786e-01 -0.06224  2.687e-01
## factor(wdicode)NLD -0.077442    0.04000 -1.93604 7.779e-02 -0.16500  1.012e-02
## factor(wdicode)NOR  0.007936    0.03838  0.20680 8.396e-01 -0.07568  9.156e-02
## factor(wdicode)NZL -0.049783    0.06086 -0.81797 4.330e-01 -0.18600  8.643e-02
## factor(wdicode)PAK -0.006541    0.07944 -0.08234 9.368e-01 -0.19652  1.834e-01
## factor(wdicode)PAN -0.088638    0.01993 -4.44643 4.954e-03 -0.13808 -3.919e-02
## factor(wdicode)PER -0.002910    0.07418 -0.03923 9.695e-01 -0.16888  1.631e-01
## factor(wdicode)PHL  0.009445    0.03798  0.24868 8.079e-01 -0.07357  9.246e-02
## factor(wdicode)POL  0.273890    0.07168  3.82117 5.562e-03  0.10717  4.406e-01
## factor(wdicode)ROU  0.084028    0.12871  0.65285 5.386e-01 -0.23268  4.007e-01
## factor(wdicode)SWE  0.191165    0.03234  5.91087 4.850e-04  0.11549  2.668e-01
## factor(wdicode)SYR  0.022376    0.04769  0.46922 6.506e-01 -0.08624  1.310e-01
## factor(wdicode)TUN -0.052829    0.05733 -0.92152 3.758e-01 -0.17842  7.277e-02
## factor(wdicode)TUR -0.112680    0.04193 -2.68742 2.074e-02 -0.20470 -2.066e-02
## factor(wdicode)UKR  0.471099    0.07341  6.41701 4.632e-05  0.30981  6.324e-01
## factor(wdicode)VEN  0.015312    0.04000  0.38285 7.122e-01 -0.07755  1.082e-01
## factor(wdicode)YUG  0.112982    0.06256  1.80606 9.638e-02 -0.02353  2.495e-01
## factor(year)1948   -0.028181    0.02863 -0.98428 3.669e-01 -0.10010  4.374e-02
## factor(year)1949   -0.055047    0.05486 -1.00347 3.392e-01 -0.17721  6.712e-02
## factor(year)1950    0.122365    0.10111  1.21024 2.542e-01 -0.10313  3.479e-01
## factor(year)1951   -0.072065    0.04762 -1.51339 1.719e-01 -0.18359  3.946e-02
## factor(year)1952   -0.048187    0.04922 -0.97893 3.660e-01 -0.16914  7.277e-02
## factor(year)1953   -0.191119    0.06327 -3.02066 2.248e-02 -0.34477 -3.747e-02
## factor(year)1954   -0.070158    0.04713 -1.48856 2.126e-01 -0.20235  6.204e-02
## factor(year)1955    0.128070    0.06591  1.94301 1.033e-01 -0.03592  2.921e-01
## factor(year)1956   -0.046305    0.07391 -0.62647 5.542e-01 -0.22744  1.348e-01
## factor(year)1957   -0.083476    0.04626 -1.80448 1.135e-01 -0.19254  2.559e-02
## factor(year)1958   -0.066850    0.03688 -1.81241 1.086e-01 -0.15234  1.864e-02
## factor(year)1959   -0.139828    0.05199 -2.68966 2.201e-02 -0.25507 -2.458e-02
## factor(year)1960   -0.070239    0.04740 -1.48181 1.830e-01 -0.18287  4.239e-02
## factor(year)1961    0.050614    0.07187  0.70422 4.993e-01 -0.11220  2.134e-01
## factor(year)1962   -0.143814    0.06788 -2.11862 7.403e-02 -0.30613  1.850e-02
## factor(year)1963    0.035907    0.04330  0.82927 4.420e-01 -0.07295  1.448e-01
## factor(year)1964   -0.085956    0.03944 -2.17926 1.100e-01 -0.20581  3.389e-02
##                        DF
## (Intercept)         7.116
## l_pctagreeus_unga  10.649
## factor(wdicode)AUS 12.197
## factor(wdicode)BEL 12.824
## factor(wdicode)BOL  2.939
## factor(wdicode)BRA 12.157
## factor(wdicode)CAN  9.788
## factor(wdicode)CHL  9.646
## factor(wdicode)CIV  2.907
## factor(wdicode)COL 13.278
## factor(wdicode)CSK  4.055
## factor(wdicode)CUB 12.371
## factor(wdicode)DNK  6.775
## factor(wdicode)ECU 11.237
## factor(wdicode)EGY 10.861
## factor(wdicode)GHA  7.704
## factor(wdicode)GRC  8.294
## factor(wdicode)IND 11.315
## factor(wdicode)IRL  5.980
## factor(wdicode)IRN  9.423
## factor(wdicode)IRQ  9.172
## factor(wdicode)ITA  5.333
## factor(wdicode)JPN  5.681
## factor(wdicode)LBN  7.129
## factor(wdicode)LBR  5.878
## factor(wdicode)LKA  8.434
## factor(wdicode)MAR  6.130
## factor(wdicode)NLD 11.514
## factor(wdicode)NOR 11.991
## factor(wdicode)NZL  9.682
## factor(wdicode)PAK  6.631
## factor(wdicode)PAN  5.683
## factor(wdicode)PER  9.710
## factor(wdicode)PHL 11.665
## factor(wdicode)POL  7.623
## factor(wdicode)ROU  5.865
## factor(wdicode)SWE  7.384
## factor(wdicode)SYR  8.613
## factor(wdicode)TUN 11.434
## factor(wdicode)TUR 11.263
## factor(wdicode)UKR 11.166
## factor(wdicode)VEN  7.697
## factor(wdicode)YUG 11.837
## factor(year)1948    5.419
## factor(year)1949   10.040
## factor(year)1950    9.931
## factor(year)1951    7.348
## factor(year)1952    5.898
## factor(year)1953    6.193
## factor(year)1954    3.900
## factor(year)1955    5.617
## factor(year)1956    5.963
## factor(year)1957    7.104
## factor(year)1958    7.772
## factor(year)1959   10.393
## factor(year)1960    6.836
## factor(year)1961    8.918
## factor(year)1962    6.636
## factor(year)1963    5.403
## factor(year)1964    3.271
##
## Multiple R-squared:  0.8851 ,    Adjusted R-squared:  0.7439
## F-statistic:    NA on 59 and 41 DF,  p-value: NA
```

```r
unsc_data_clean <- unsc_data %>%
  filter(complete.cases(.))

stan_data <- list(
  N = nrow(unsc_data_clean),
  C = length(unique(unsc_data_clean$wdicode)),
  Y = length(unique(unsc_data_clean$year)),
  country = as.integer(factor(unsc_data_clean$wdicode)),
  year = as.integer(factor(unsc_data_clean$year)),
  l_pctagreeus_unga = unsc_data_clean$l_pctagreeus_unga,
  DependentVariable = unsc_data_clean$DependentVariable
)

# Fit the model
fit <- stan(
  file = "RandomIntercepts.stan",
  data = stan_data,
  warmup = 500,
  iter = 2000,
  chains = 2,
  seed = 42
)
```

5. Now fit a Bayesian linear model with random intercepts (not slopes) for each country and year. Be sure to carry out appropriate diagnostics. Compare your results to those above. (Note: You may end up with more countries in this analysis than in problem 4. Countries that only appear in the dataset once may be dropped from the fixed effects model depending on how you implemented it.)

```
## Trying to compile a simple C file

## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
## using C compiler: ‘Apple clang version 15.0.0 (clang-1500.3.9.4)’
## using SDK: ‘MacOSX14.4.sdk’
## clang -arch arm64 -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG   -I"/Library/Framew
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/library/StanHeade
## In file included from /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/library/RcppEigen/
## In file included from /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/library/RcppEigen/
## /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/library/RcppEigen/include/Eigen/src/Core
## #include <cmath>
##          ^~~~~~~
## 1 error generated.
## make: *** [foo.o] Error 1
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 7e-05 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.7 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  501 / 2000 [ 25%]  (Sampling)
## Chain 1: Iteration:  700 / 2000 [ 35%]  (Sampling)
## Chain 1: Iteration:  900 / 2000 [ 45%]  (Sampling)
## Chain 1: Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 1: Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 1: Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 1: Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 1: Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
```

```
## Chain 1:
## Chain 1:  Elapsed Time: 0.095 seconds (Warm-up)
## Chain 1:                0.281 seconds (Sampling)
## Chain 1:                0.376 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 7e-06 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.07 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  501 / 2000 [ 25%]  (Sampling)
## Chain 2: Iteration:  700 / 2000 [ 35%]  (Sampling)
## Chain 2: Iteration:  900 / 2000 [ 45%]  (Sampling)
## Chain 2: Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 2: Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 2: Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 2: Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 2: Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.109 seconds (Warm-up)
## Chain 2:                0.287 seconds (Sampling)
## Chain 2:                0.396 seconds (Total)
## Chain 2:
```

When comparing the models, I will specifically focus on countries and years, and compare at the statistically significant level of $\alpha < 0.05$ for the fixed effects model, and generally look at 95% credible interval for the bayesian approach.

From a country perspective, the fixed effects model highlights a few countries as being statistically significant from the base country (Argentina) in their voting against the US in the UNSC. The countries that were statistically significant with the largest effect sizes were Checkoslavakia, Lebanon, India, Poland, Sweden and Ukraine. Among those Poland was the only one that had large negative coefficients indicating it is more likely to vote with the US. Compared to the bayesian model, Poland (alpha_country[33]) has a 95% credible interval that spans [-0.09, 0.13], showing that there is no definitive likelihood with Poland voting for or against US. Among the others with positive coefficients, the countries that showed consistent effect sizes between models were Ukraine (alpha_country[40]) and India (alpha_country[17]), indicating that these countries were more likely to vote against the US in the UNSC.

From the years perspective, the fixed effects model highlights 1953 as being significantly different from the base year of 1965. When I compare this to the stan model, the year 1953 (alpha_year[7]) has a 95% credible interval that spans [-0.19, -0.001]. Here we see some consistency between the models, as the fixed effects model insuated that the likelihood of a country voting against the US in the UNSC was significantly lower than in the baseline year, and the bayesian credible interval showed the statistic credible interval to be negative as well. The consistency makes this insuation seem more conclusive.

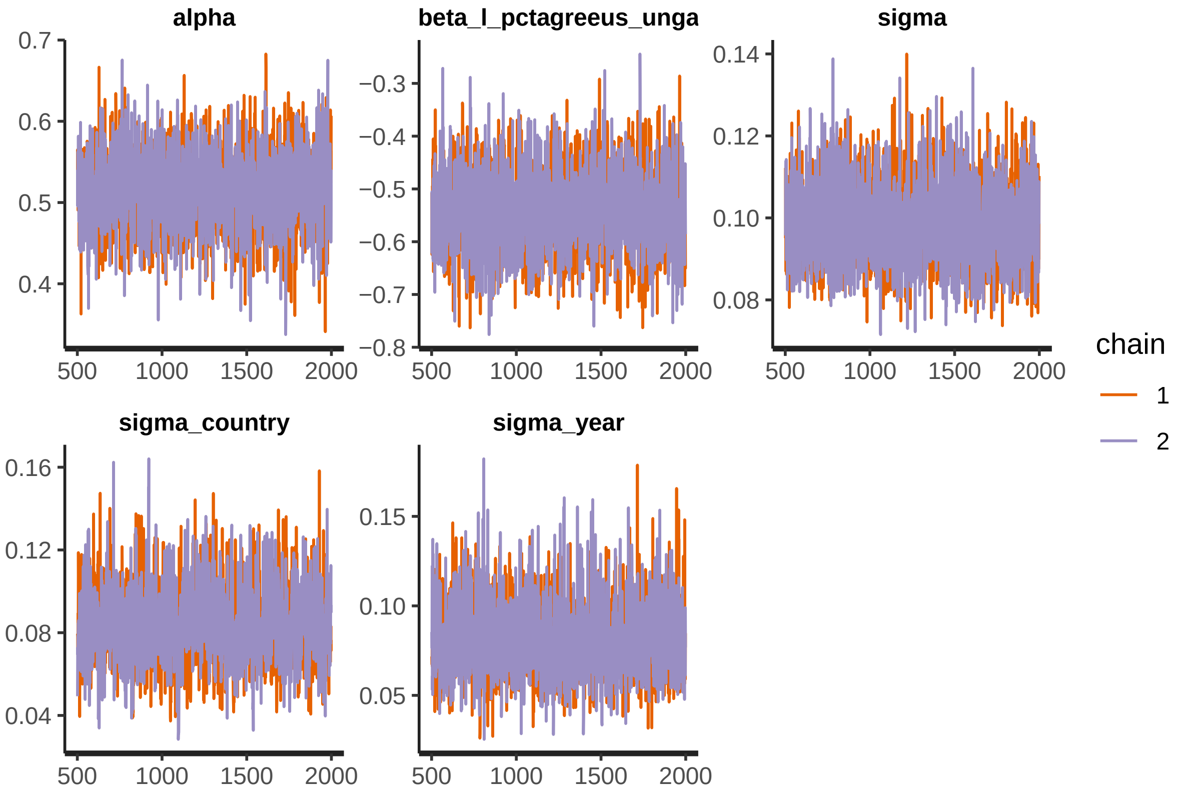
One additional note is regarding Lagged percent ideological agreemenent with the USA. Both models showed a negative association compared to the Dependent variable, but the fixed effects model had a p-vale ~ 0.12. So the evidence that it is a negative association is weak, but arguable since it does show a credible interval of [-0.68, -0.38], which indicates a large effect size.

In addition, the traceplot, effective sample size, and Rhat below show that the stan model converged, allowing for reliable diagnostics.

```
print(fit, pars = c("alpha", "beta_l_pctagreeus_unga", "sigma", "sigma_country", "sigma_year"))
```

```
## Inference for Stan model: anon_model.
## 2 chains, each with iter=2000; warmup=500; thin=1;
## post-warmup draws per chain=1500, total post-warmup draws=3000.
##
##                         mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff
## alpha                   0.52       0 0.05  0.42  0.48  0.52  0.55  0.61  1721
## beta_l_pctagreeus_unga -0.54       0 0.07 -0.68 -0.59 -0.54 -0.49 -0.39  1610
## sigma                   0.10       0 0.01  0.08  0.09  0.10  0.11  0.12  2054
## sigma_country           0.08       0 0.02  0.05  0.07  0.08  0.10  0.12  1082
## sigma_year              0.08       0 0.02  0.05  0.07  0.08  0.09  0.13  2203
##                        Rhat
## alpha                     1
## beta_l_pctagreeus_unga    1
## sigma                     1
## sigma_country             1
## sigma_year                1
##
## Samples were drawn using NUTS(diag_e) at Tue Dec 17 17:20:45 2024.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
traceplot(fit, pars = c("alpha", "beta_l_pctagreeus_unga", "sigma", "sigma_country", "sigma_year"))
```



```r
print(summary(fit)$summary)
```

```
##                                 mean      se_mean          sd          2.5%
## alpha                    0.515087477 0.0011274256 0.046770038  4.221713e-01
## alpha_country[1]         0.007577597 0.0007128915 0.046634998 -8.669802e-02
## alpha_country[2]         0.001975331 0.0007236445 0.049582541 -9.274957e-02
## alpha_country[3]        -0.057698830 0.0007753074 0.046055105 -1.454664e-01
## alpha_country[4]        -0.028487234 0.0009420791 0.064932088 -1.622537e-01
## alpha_country[5]        -0.028823695 0.0007104146 0.039506635 -1.085514e-01
## alpha_country[6]         0.005134249 0.0007076094 0.048323694 -9.089098e-02
## alpha_country[7]        -0.041547255 0.0007593844 0.047017157 -1.346208e-01
## alpha_country[8]        -0.015848353 0.0009349683 0.065347090 -1.458361e-01
## alpha_country[9]        -0.075148219 0.0007439948 0.042129286 -1.606314e-01
## alpha_country[10]        0.067742814 0.0011734816 0.070097946 -6.610236e-02
## alpha_country[11]       -0.042438287 0.0007831116 0.048746180 -1.383467e-01
## alpha_country[12]       -0.015482741 0.0008589647 0.058380768 -1.365005e-01
## alpha_country[13]       -0.075845850 0.0008537536 0.048987035 -1.710200e-01
## alpha_country[14]       -0.015277458 0.0006963896 0.046843827 -1.086017e-01
## alpha_country[15]       -0.064051699 0.0010265803 0.057344863 -1.820750e-01
## alpha_country[16]       -0.061527811 0.0010559958 0.058610540 -1.770795e-01
## alpha_country[17]        0.116258265 0.0012145906 0.062161875  4.468208e-04
## alpha_country[18]        0.007026937 0.0009194713 0.066958445 -1.234042e-01
## alpha_country[19]       -0.046091717 0.0008569475 0.058187490 -1.664902e-01
## alpha_country[20]       -0.030748004 0.0008695457 0.056837159 -1.469465e-01
## alpha_country[21]       -0.026638594 0.0008594303 0.056932284 -1.410377e-01
## alpha_country[22]       -0.004666332 0.0008826052 0.058864135 -1.218196e-01
## alpha_country[23]        0.099641226 0.0011351652 0.060829790 -1.803112e-02
## alpha_country[24]        0.019820624 0.0009056575 0.063603431 -1.044454e-01
## alpha_country[25]        0.029944840 0.0008748156 0.056021819 -7.989256e-02
## alpha_country[26]        0.009694269 0.0009058565 0.057273678 -1.005473e-01
## alpha_country[27]       -0.014221798 0.0008886066 0.056707288 -1.276830e-01
## alpha_country[28]        0.048585184 0.0007831283 0.046335046 -4.144593e-02
## alpha_country[29]       -0.015578593 0.0007971530 0.057464020 -1.312665e-01
## alpha_country[30]       -0.056042789 0.0009483588 0.059006578 -1.716001e-01
## alpha_country[31]       -0.041609032 0.0009678937 0.057222418 -1.563143e-01
## alpha_country[32]        0.030908797 0.0007945251 0.057149217 -7.846202e-02
## alpha_country[33]        0.019715167 0.0007720961 0.058301719 -9.223004e-02
## alpha_country[34]        0.096266373 0.0012116098 0.062156029 -1.957925e-02
## alpha_country[35]       -0.026251971 0.0010216867 0.067452727 -1.572931e-01
## alpha_country[36]        0.112736896 0.0009982967 0.059725150 -1.705402e-03
## alpha_country[37]       -0.024624580 0.0008001749 0.057461434 -1.364262e-01
## alpha_country[38]       -0.075563843 0.0009412990 0.057303592 -1.898134e-01
## alpha_country[39]       -0.032057092 0.0008024929 0.044915307 -1.206014e-01
## alpha_country[40]        0.192281997 0.0019403333 0.077094277  5.120241e-02
## alpha_country[41]        0.021960431 0.0008547132 0.055446774 -8.340251e-02
## alpha_country[42]        0.029434196 0.0008151608 0.052110586 -6.818799e-02
## alpha_year[1]            0.031448477 0.0006729623 0.041979538 -5.092682e-02
## alpha_year[2]            0.040080087 0.0007786810 0.042379679 -4.009627e-02
## alpha_year[3]            0.014738351 0.0007147892 0.041438733 -6.444248e-02
## alpha_year[4]            0.128115774 0.0009195704 0.045479306  4.006046e-02
## alpha_year[5]           -0.008029723 0.0007054645 0.042734018 -9.287460e-02
## alpha_year[6]           -0.035382885 0.0007402053 0.043302494 -1.266388e-01
## alpha_year[7]           -0.097584504 0.0008734107 0.045943727 -1.919322e-01
## alpha_year[8]           -0.040605876 0.0007376513 0.042556470 -1.241809e-01
## alpha_year[9]            0.079293274 0.0007429146 0.044886530 -8.053361e-03
## alpha_year[10]          -0.047315037 0.0007547268 0.043956276 -1.361482e-01
## alpha_year[11]           0.021291971 0.0009153561 0.044874300 -6.689382e-02
## alpha_year[12]           0.005965109 0.0007340059 0.043242494 -8.010819e-02
## alpha_year[13]          -0.041899680 0.0008015247 0.044598810 -1.310768e-01
## alpha_year[14]          -0.025644917 0.0007473718 0.043562363 -1.144165e-01
## alpha_year[15]           0.057152557 0.0006899847 0.042499653 -2.531157e-02
## alpha_year[16]          -0.089931891 0.0008618367 0.045380815 -1.815490e-01
## alpha_year[17]           0.043264504 0.0007706791 0.042937539 -3.875942e-02
## alpha_year[18]          -0.045026082 0.0007804855 0.044557221 -1.328747e-01
## beta_l_pctagreeus_unga  -0.536487855 0.0018499530 0.074239461 -6.827701e-01
## sigma                    0.098944383 0.0002127965 0.009645225  8.137005e-02
## sigma_country            0.083463882 0.0005588079 0.018382511  4.905214e-02
## sigma_year               0.080601093 0.0004356722 0.020447614  4.647886e-02
## lp__                   310.043461195 0.2660702766 7.437059925  2.945666e+02
##                                  25%           50%           75%         97.5%
## alpha                    0.483932992   0.516362468   0.546713510   0.606652528
## alpha_country[1]        -0.022521579   0.007465750   0.039177017   0.097882179
## alpha_country[2]        -0.031652276   0.002024796   0.035887295   0.099741446
## alpha_country[3]        -0.089405107  -0.057031343  -0.025797236   0.032591446
## alpha_country[4]        -0.069757432  -0.027533964   0.013717574   0.096780593
## alpha_country[5]        -0.054552281  -0.028422125  -0.002416553   0.048490666
## alpha_country[6]        -0.027073849   0.005373445   0.037289906   0.097694343
## alpha_country[7]        -0.073087818  -0.040963027  -0.008540866   0.047214421
## alpha_country[8]        -0.058474492  -0.014661788   0.027140423   0.109538100
## alpha_country[9]          -0.102414350  -0.074859578  -0.047066863   0.005509301
## alpha_country[10]          0.020539229   0.064059590   0.115067933   0.211529317
## alpha_country[11]         -0.075293538  -0.042347523  -0.009307638   0.051537662
## alpha_country[12]         -0.052045549  -0.014575951   0.021592446   0.101587378
## alpha_country[13]         -0.108735950  -0.076032658  -0.043439429   0.021724787
## alpha_country[14]         -0.046769633  -0.014804518   0.015658585   0.078358138
## alpha_country[15]         -0.101148994  -0.062784283  -0.025159200   0.048426112
## alpha_country[16]         -0.101780658  -0.061258548  -0.020732962   0.047835779
## alpha_country[17]          0.072803930   0.115662280   0.158172352   0.239853681
## alpha_country[18]         -0.035947561   0.006950010   0.049443695   0.141961696
## alpha_country[19]         -0.083968934  -0.043250107  -0.006520154   0.064345095
## alpha_country[20]         -0.068282067  -0.029522767   0.007615622   0.080648834
## alpha_country[21]         -0.063362087  -0.025234445   0.011063942   0.083331378
## alpha_country[22]         -0.043140869  -0.003932276   0.034202150   0.111040769
## alpha_country[23]          0.058505059   0.097417061   0.139326277   0.223350288
## alpha_country[24]         -0.024325938   0.020466209   0.063782125   0.141397309
## alpha_country[25]         -0.007634726   0.029508529   0.066563221   0.142183519
## alpha_country[26]         -0.028267217   0.009156009   0.046832262   0.123596856
## alpha_country[27]         -0.051145459  -0.011862250   0.023685042   0.094755375
## alpha_country[28]          0.016801547   0.048378353   0.079021432   0.139535579
## alpha_country[29]         -0.053035178  -0.014041769   0.021407227   0.094523092
## alpha_country[30]         -0.094789735  -0.057078042  -0.016802836   0.060244921
## alpha_country[31]         -0.080030483  -0.039746604  -0.003216807   0.067367884
## alpha_country[32]         -0.006698598   0.029046884   0.068553700   0.146355741
## alpha_country[33]         -0.020782232   0.019107253   0.060297354   0.132912064
## alpha_country[34]          0.052801695   0.094424808   0.135606952   0.224812834
## alpha_country[35]         -0.070154973  -0.026875758   0.017534022   0.110399648
## alpha_country[36]          0.072505410   0.112647876   0.152608694   0.233541924
## alpha_country[37]         -0.062856524  -0.023548094   0.013081719   0.086319283
## alpha_country[38]         -0.115089408  -0.075484907  -0.035826547   0.034133667
## alpha_country[39]         -0.061727601  -0.032176696  -0.001231448   0.055470559
## alpha_country[40]          0.138010768   0.190710017   0.242658367   0.348810680
## alpha_country[41]         -0.014928274   0.021242082   0.060194432   0.134164653
## alpha_country[42]         -0.006101936   0.028877165   0.063845841   0.133729567
## alpha_year[1]              0.003089598   0.031844094   0.058680659   0.114375063
## alpha_year[2]              0.011533841   0.038204939   0.066802090   0.125710842
## alpha_year[3]             -0.013275062   0.013391198   0.042815191   0.097211729
## alpha_year[4]              0.096397407   0.126831208   0.158445214   0.219586198
## alpha_year[5]             -0.035882773  -0.007705790   0.021061198   0.076082279
## alpha_year[6]             -0.063455515  -0.034727146  -0.005512157   0.046274012
## alpha_year[7]             -0.128192521  -0.096017163  -0.066257853  -0.009889082
## alpha_year[8]             -0.069716821  -0.040526338  -0.011289337   0.039782392
## alpha_year[9]              0.049119188   0.079930363   0.109311761   0.165993144
## alpha_year[10]            -0.077465985  -0.047172442  -0.017201568   0.036607535
## alpha_year[11]            -0.007942445   0.021358754   0.051752040   0.111794249
## alpha_year[12]            -0.023831685   0.005769432   0.035517427   0.088518727
## alpha_year[13]            -0.073007364  -0.039859336  -0.011639764   0.040153802
## alpha_year[14]            -0.054438624  -0.024242173   0.003656353   0.055935668
## alpha_year[15]             0.029279881   0.056861295   0.084962691   0.141135231
## alpha_year[16]            -0.119676803  -0.089026369  -0.059785901  -0.003654300
## alpha_year[17]             0.013500432   0.043125198   0.072004916   0.125931307
## alpha_year[18]            -0.073278731  -0.045333777  -0.014739594   0.039516441
## beta_l_pctagreeus_unga    -0.585848522  -0.537193143  -0.487619622  -0.388444450
## sigma                      0.092175765   0.098410900   0.105343380   0.118861776
## sigma_country             0.071036947   0.082870080   0.095013763   0.123534518
## sigma_year                0.066073567   0.078615883   0.092243048   0.127512993
## lp__                    305.220928685 310.287430548 315.186037058 323.704788312
##                            n_eff      Rhat
## alpha                  1720.9161 0.9997277
## alpha_country[1]       4279.3426 0.9996236
## alpha_country[2]       4694.6911 0.9994188
## alpha_country[3]       3528.6423 0.9995568
## alpha_country[4]       4750.5510 0.9997315
## alpha_country[5]       3092.5474 0.9999308
## alpha_country[6]       4663.7270 0.9998186
## alpha_country[7]       3833.4459 0.9996201
## alpha_country[8]       4884.9342 0.9995023
## alpha_country[9]       3206.4791 0.9997059
## alpha_country[10]      3568.2726 1.0001973
## alpha_country[11]      3874.6622 0.9994644
## alpha_country[12]      4619.4362 0.9993730
## alpha_country[13]      3292.2834 1.0010994
## alpha_country[14]      4524.8084 0.9998130
## alpha_country[15]      3120.3493 1.0000867
## alpha_country[16]      3080.5417 0.9999977
## alpha_country[17]      2619.3189 0.9996712
## alpha_country[18]      5303.1557 0.9993758
## alpha_country[19]      4610.5295 0.9999372
## alpha_country[20]      4272.4790 0.9997286
## alpha_country[21]      4388.2957 1.0003404
## alpha_country[22]      4448.0388 1.0000988
## alpha_country[23]      2871.5377 1.0014459
## alpha_country[24]      4932.1139 0.9997083
## alpha_country[25]      4100.9206 0.9997096
## alpha_country[26]      3997.5261 0.9998411
## alpha_country[27]      4072.4774 0.9994031
## alpha_country[28]      3500.6877 0.9994786
## alpha_country[29]      5196.4731 1.0002937
## alpha_country[30]      3871.2880 0.9997536
## alpha_country[31]      3495.2407 0.9998522
## alpha_country[32]      5173.7485 0.9996459
## alpha_country[33]      5701.9049 0.9996232
## alpha_country[34]      2631.7279 1.0000771
## alpha_country[35]      4358.7659 0.9993985
## alpha_country[36]      3579.2762 0.9994300
## alpha_country[37]      5156.8337 0.9994023
## alpha_country[38]      3706.0266 1.0005490
## alpha_country[39]      3132.6105 0.9998489
## alpha_country[40]      1578.6709 1.0006937
## alpha_country[41]      4208.3486 0.9994288
## alpha_country[42]      4086.6300 0.9998432
## alpha_year[1]          3891.2926 0.9993552
## alpha_year[2]          2962.0757 0.9997329
## alpha_year[3]          3360.9106 0.9995599
## alpha_year[4]          2446.0070 0.9994779
## alpha_year[5]          3669.4173 0.9999287
## alpha_year[6]          3422.3263 0.9998822
## alpha_year[7]          2767.0398 0.9999314
## alpha_year[8]          3328.3494 1.0005894
## alpha_year[9]          3650.5162 0.9995735
## alpha_year[10]         3392.0497 0.9995226
## alpha_year[11]         2403.3402 1.0001677
## alpha_year[12]         3470.7425 0.9994329
## alpha_year[13]         3096.0840 0.9995310
## alpha_year[14]         3397.4217 1.0005381
## alpha_year[15]         3793.9516 0.9997255
## alpha_year[16]         2772.6474 0.9993982
## alpha_year[17]         3104.0384 0.9998646
## alpha_year[18]         3259.1661 0.9995691
## beta_l_pctagreeus_unga 1610.4536 0.9999531
## sigma                  2054.4503 1.0004405
## sigma_country          1082.1428 1.0005703
## sigma_year             2202.7524 0.9994360
## lp__                    781.2856 0.9994629
```