BDA Project 2021 — Bayesian analysis of trends in IQ test scores over successive generations

Anonymous

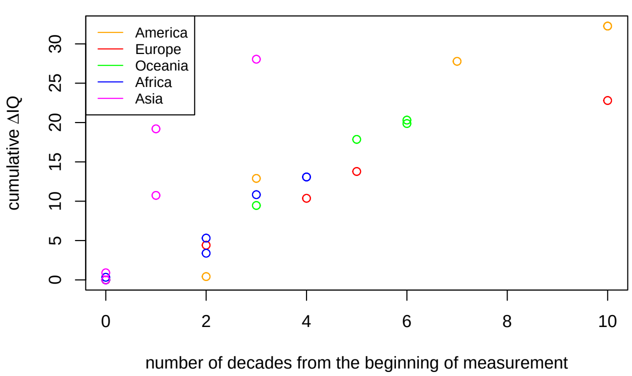
6.12.2021

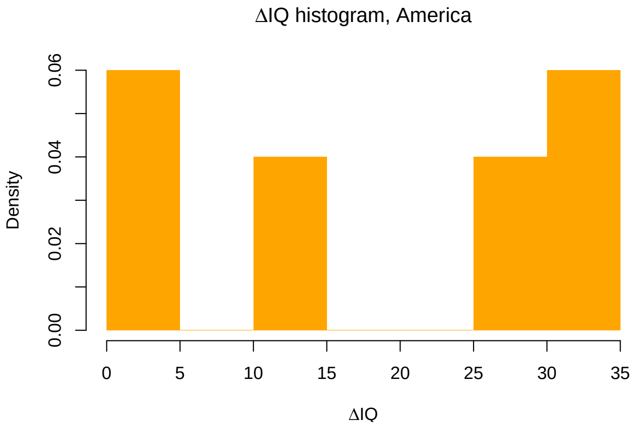
Introduction

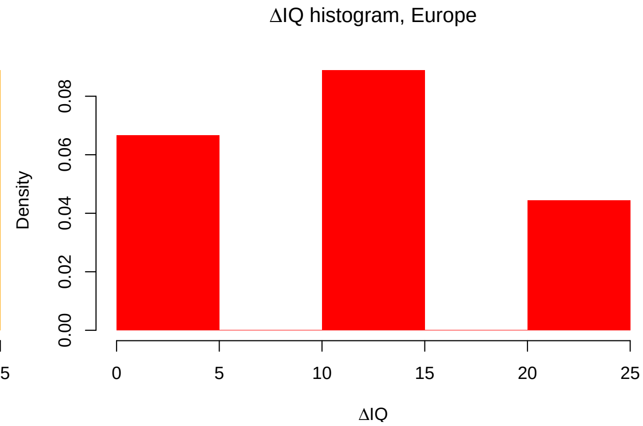
IQ is a psychometric developed in 1904 by French Psychologsts Alfred Binet and Theodore Simon as a measurement intended to measure the test taker's cognitive ability. IQ is by convention normally distributed with a mean of 100 points and a standard deviation of 15 points. Full-scale IQ is an IQ score derived by averaging over the results of a test battery of several subtests which are intended to measure different intelligence domains. IQ tests are not practiced for and as such are supposed to measure some inherent feature of the test taker. IQ as a metric has been widely criticized, due to its spurious relation to an ordinary language understanding of intelligence and historically grave misuse of results, but remains in wide use. The observed global trend of the average IQ to rise over generations has been termed the Flynn effect. (Pietschnig and Voracek 2015). This report does not assume any position on the validity of IQ as a metric.

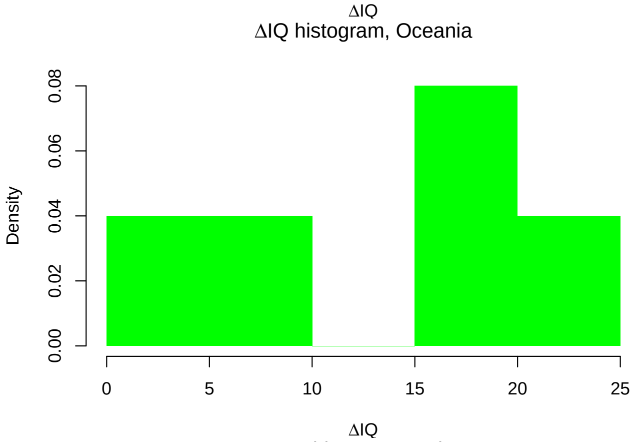
In this report we will attempt to replicate the Flynn effect with our data and to explore the dependency between IQ change and change in average years of schooling in the population (Prados de la Escosura 2021a). We will explore these dependencies both globally and per continent. The trend in IQ will be validated with a model that explores temporal dependency alone and a further model that factors in schooling level. We will model the data with a pooled, separate and hierarchical model. The models will explore the relationship between the passage of time and the development of IQ scores from the start of testing and the relationship between schooling level and the development of IQ scores from the start of testing.

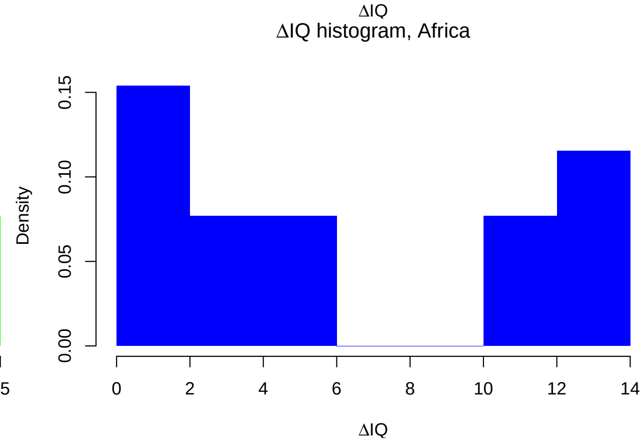
The raw data can be visualized as follows with a scatterplot of cumulative Full-scale IQ change with respect to time, a histogram of the distribution of this IQ change per continent, and histograms to show the distribution of schooling level:

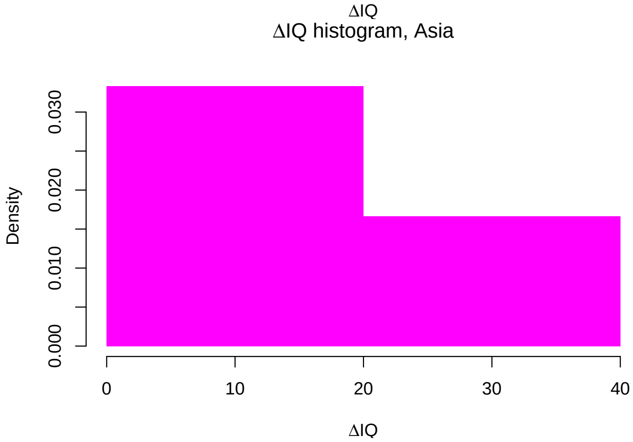


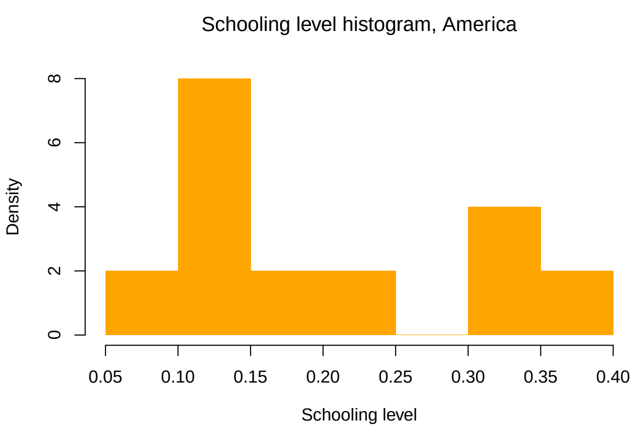

ΔIQ histogram, America
Density
ΔIQ

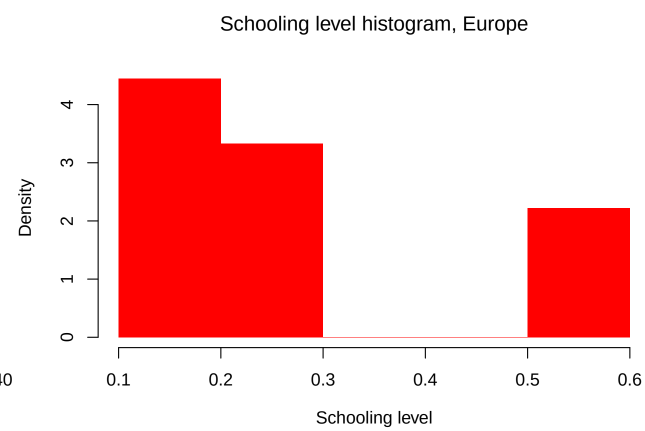

ΔIQ histogram, Europe
Density
ΔIQ

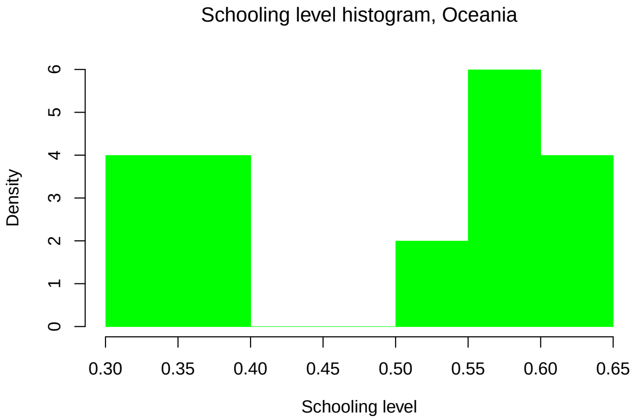

ΔIQ histogram, Oceania
Density
ΔIQ

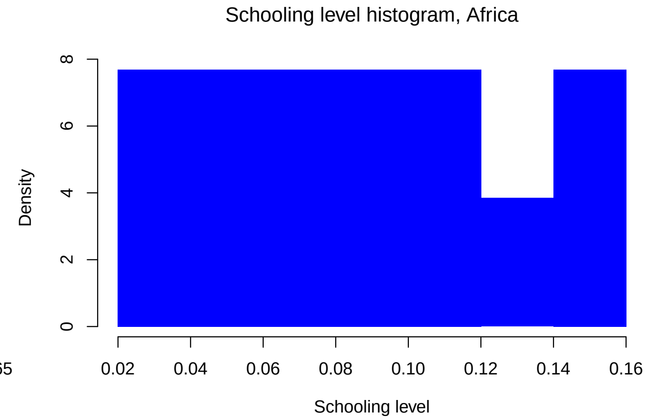

ΔIQ histogram, Africa
Density
ΔIQ

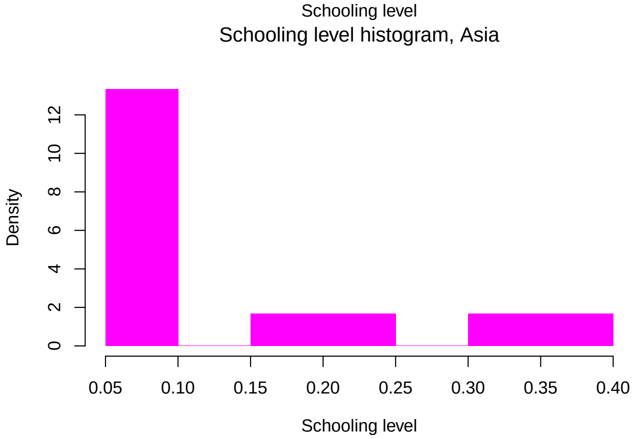

ΔIQ histogram, Asia
Density
ΔIQ


Schooling level histogram, America
Density
Schooling level


Schooling level histogram, Europe
Density
Schooling level







Data description

Our data is a collection of time indexed IQ test score changes for a given continents (Asia, Europe, Oceania, Africa, America). Successive data points are the developments in IQ as compared to the first measurement made, such that at the first Fullscale IQ Change is zero. Schooling levels are not set to be zero for the first data point but this didn't seem to cause problems for the models.

The Full-scale IQ data has previously been used in a frequentist meta-analysis which gathered global historical IQ data from 1909 to 2013 and found global non-linear increases across all IQ domains. Best explanation for IQ increases was found in increasing nutrition, education and lower family size. (Pietschnig and Voracek 2015).

Our analysis differs in that it is bayesian and studies the relationship between education level and IQ directly. This approach is vulnerable to overfitting assuming education level is determined by for example income level but this is not a concern for us as we are interested in the link between education level and IQ increase specifically. Schooling level or time could also both explain both of the models if they are dependent on each other, as they can be assumed to be. The joint models which take into account both temporal change and schooling level, were they to perform better than the separate models, would reveal a d

The education level data has been analysed before with a trend analysis from 1870 onwards discovering gains globally that do not match global wealth distribution such that the global middle class has seen the best gains overall. (Prados de la Escosura 2021a).

Our analysis is bayesian and concerned with the link to IQ which has not previously been explored for this data set.

- Full-scale IQ Data acquired from (OWID 2018), preprocessing discussed in Appendix A: Preprocessing

- Education data acquired from (Prados de la Escosura 2021b), preprocessing discussed in Appendix A: Preprocessing)

Model description

The data is organized into continents so we will model the difference and similarity in continent behavior with a separate, pooled and hierarchical model.

The pooled model will pool all results into a global pool for which we will simulate the following models

$$\begin{aligned}
1 &: \Delta IQ \sim N(\alpha + \beta \cdot \text{decade}, 1) \\
2 &: \Delta IQ \sim N(\alpha + \gamma \cdot \text{schooling level}, 1) \\
3 &: \Delta IQ \sim N(\alpha + \beta \cdot \text{decade} + \gamma \cdot \text{schooling level}, 1)
\end{aligned}$$

Each model will use weakly informative priors such that

$$\begin{aligned}
\alpha &\sim N(0, 1) \\
\beta &\sim N(0, 50) \\
\gamma &\sim N(0, 100)
\end{aligned}$$

These priors are based on our prior beliefs as follows: + The model should naturally intercept at zero, since it is modelling change to the initial state. The intercept is normally distributed to better detect unexpected behavior in the model. The variance is quite strict since the maximum change in IQ in the data is around 40 points but this should not matter since the model is not concerned with modelling anything at $t = 0$. + The temporal slope β, a slope that expresses the dependence between the monotonically increasing decade attribute and the change in IQ, should be allowed to, in terms of the prior distribution, explain all the change in the data and allowing for changes outside of the data. Our prior beliefs dictate that seeing a 100 point change would be very very unlikely but still allowed in terms of probabilities. There's conceptually an over-fitting to the data here but a more dispersed prior would be very strange rationally i.e. a decade resulting in a 100 IQ point change given that the maximum IQ score is 200. + The schooling slope, γ, that expresses the dependence between IQ change and schooling level is chosen with the same rationale as the slope β expect scaled to fit the smaller data values (schooling level values vary between 0 and 1 such that 1 implies university level education). + y's variance, the noise, is distributed normally and somewhat strictly due to us not wanting for much of the variance to be explained by noise, believing that a stricter noise would result in us detecting bad fits for our models more quickly.

The hierarchical model will have continents sharing variance for the slopes and intercepts such that for continent $c \in C$

$$\begin{aligned}
\alpha_c &\sim N(0, 1) \\
\beta_c &\sim N(0, 50) \\
\gamma_c &\sim N(0, 100) \\
\sigma &\sim N(0, 100)
\end{aligned}$$

Where σ is the shared variance between continents for the linear combination of attributes. In total, the hierarchical model is then constructed such that

$$\Delta IQ_c \sim N(\alpha_c + \beta_c \cdot \text{decade} + \gamma_c \cdot \text{schooling level}, \sigma)$$

Stan code

The Stan code was run using rstan with 4 chains, 4000 iterations and with warm up length 2000. The final results were generated by running the code on https://jupyter.cs.aalto.fi/ servers.

Due to its length we have deciced to place the Stan source code in Appendix B: Stan source code

Convergence diagnostics

		Europe	Oceania	Africa	Asia	America
Separate simple						
$\widehat{R}$	a	1.00118	1.002355	1.000308	1.003236	1.000922
	b	1.00118	1.002355	1.000308	1.003236	1.000922
n_{eff}	a	2463	2772	3051	4591	3711
	b	2502	2863	2837	3584	2740
No divergence.						
Maximum tree depth of 10 never saturated						
Separate with Schooling Index						
$\widehat{R}$	a	1.00961	1.003344	1.001653	1.000237	1.003206
	c	1.000961	1.003344	1.001653	1.000237	1.003206
n_{eff}	a	3866	2796	2943	4826	4767
	c	3238	3414	3080	2644	3006
No divergence.						
Maximum tree depth of 10 never saturated						
Separate with Decade and Schooling Index						
$\widehat{R}$	a	1.0013204	1.0003936	1.0010471	1.0005874	0.996429
	b	1.0013204	1.0003936	1.0010471	1.0005874	0.996429
	c	1.001945	1.001503	1.001095	1.00122	1.00092
n_{eff}	a	2491	2362	2676	5330	5337
	b	2058	1989	2270	1999	2199
	c	1896	1862	2107	1982	2163
No divergence.						
Maximum tree depth of 10 never saturated						

Table 1: Convergence diagnostics for the separate model

		Europe	Oceania	Africa	Asia	America
Hierarchical temporal						
$\widehat{R}$	a	1.000584	1 .0012687	0.9999413	1.0000693	0.9999622
	b	0.9998691	1.0000395	1.0008906	1.0002912	1.0005587
n_{eff}	a	4468	4302	6278	5025	4971
	b	4809	4902	5227	4964	5097
No divergence.						
Maximum tree depth of 10 never saturated						
Hierarchical with Schooling Index						
$\widehat{R}$	a	1.000805	1.001127	1.000538	1.000633	1.000593
	c	1.0001844	1.0010336	1.0005288	1.0001078	0.9999031
n_{eff}	a	4085	4013	4356	4096	4373
	c	3531	3748	4119	3796	4230
No divergence.						
Maximum tree depth of 10 never saturated						
Hierarchical with Decade and Schooling Index						
$\widehat{R}$	a	1.000985	1.001348	1.002005	1.002113	1.000064
	b	1.0009266	0.9996498	1.0015432	1.0014392	1.0006435
	c	1.0010106	0.9994411	1.0020594	1.0014008	1.0005017
n_{eff}	a	5330	5317	4493	3833	5855
	b	3490	3542	2998	3012	3037
	c	3353	3421	2884	2795	2970
No divergence.						
Maximum tree depth of 10 never saturated						

Table 2: Convergence diagnostics for the hierarchical model

	$\widehat{R}$	n_{eff}
Pooled temporal		
a	1.001166	2719
b	1.001959	2596
No divergence.		
Maximum tree depth of 10 never saturated		
Pooled with Schooling Index		
a	1.000503	2858
b	1.001412	2603
No divergence.		
Maximum tree depth of 10 never saturated		
Pooled with Decade and Schooling Index		
a	1.001149	2994
b	1.001361	2195
c	1.000848	2057
No divergence.		
Maximum tree depth of 10 never saturated		

Table 3: Convergence diagnostics for the pooled model

$\widehat{\mathbf{R}}$ is roughly the ratio of the parameter variance when the data is pooled across all of the chains to the within-chain variance. It basically measures how much the chains are reaching different conclusions. Since most of the Rhats above are very close to 1, and all of them are less than 1.1, we can conclude that the chains reach the same conclusion.

$\mathbf{n_{eff}}$ is an estimate of the effective sample size of the parameters. The smaller it is, the greater is the uncertainty associated with its parameter. A common heuristic is that the nEff needs to be higher than 100 for parameters of interest. Since this is the case with the above models, we can be sure that there is low uncertainty associated with the model parameters.

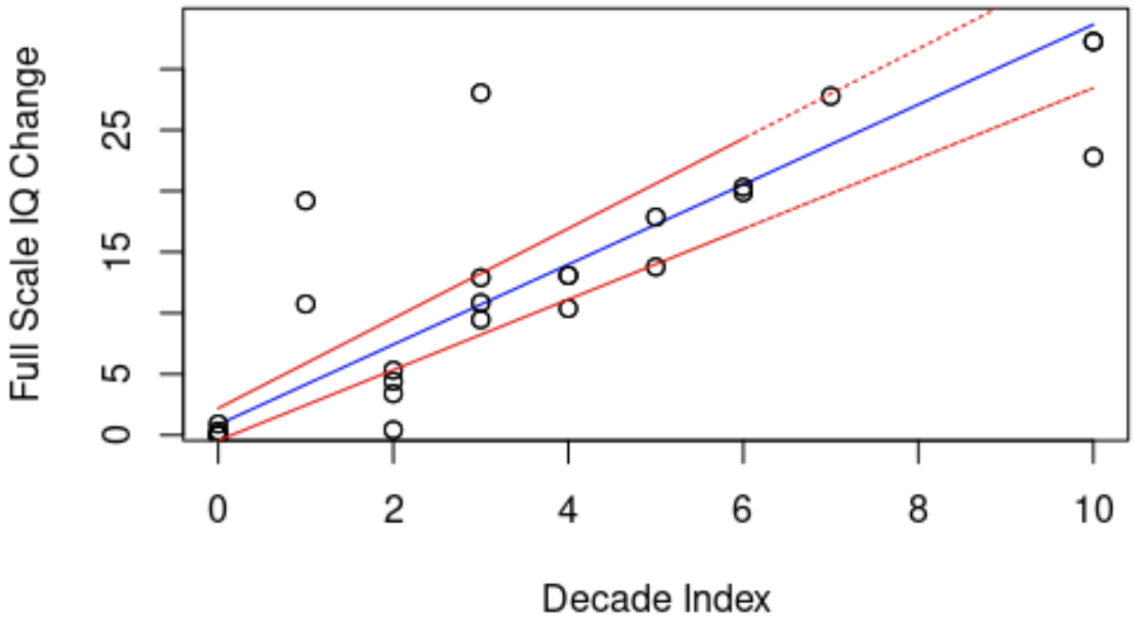
Divergence implies that the drawn samples are not from the entire posterior and thus the inferences will be biased. Since the above models do not diverge, we can conclude that they sample from the entire posterior.

Crossing the maximum **tree depth** implies that the optimal number of steps for each iteration is higher than the currently set maximum. Since this is not the case in the above models, we can conclude that the sampler is efficient and the optimal number of steps for each iteration is less than the maximum tree depth.

Performance assessment

Pooled Simple	
PSIS-LOO	-179.39
p_{eff}	3.32
Pooled with Schooling	
PSIS-LOO	-200.23
p_{eff}	2.07
Pooled with Decade and Schooling	
PSIS-LOO	-180.03
p_{eff}	3.45
Separate Simple	
PSIS-LOO	-86865.18
p_{eff}	83215.36
Separate with Schooling	
PSIS-LOO	-99097.46
p_{eff}	96147.08
Separate with Decade and Schooling	
PSIS-LOO	-99097.5
p_{eff}96147.1	
Hierarchical Simple	
PSIS-LOO	-67709.76
p_{eff}	15522.70
Hierarchical with Schooling	
PSIS-LOO	-67381.48
p_{eff}	15453.77
Separate with Decade and Schooling	
PSIS-LOO	-67748.59
p_{eff}	14991.61

The best performing model with only the decade index created the following plot. The plot includes both the expected values of the intercept and the slope, but also the lines depicting the bounds for the 90% credible in-

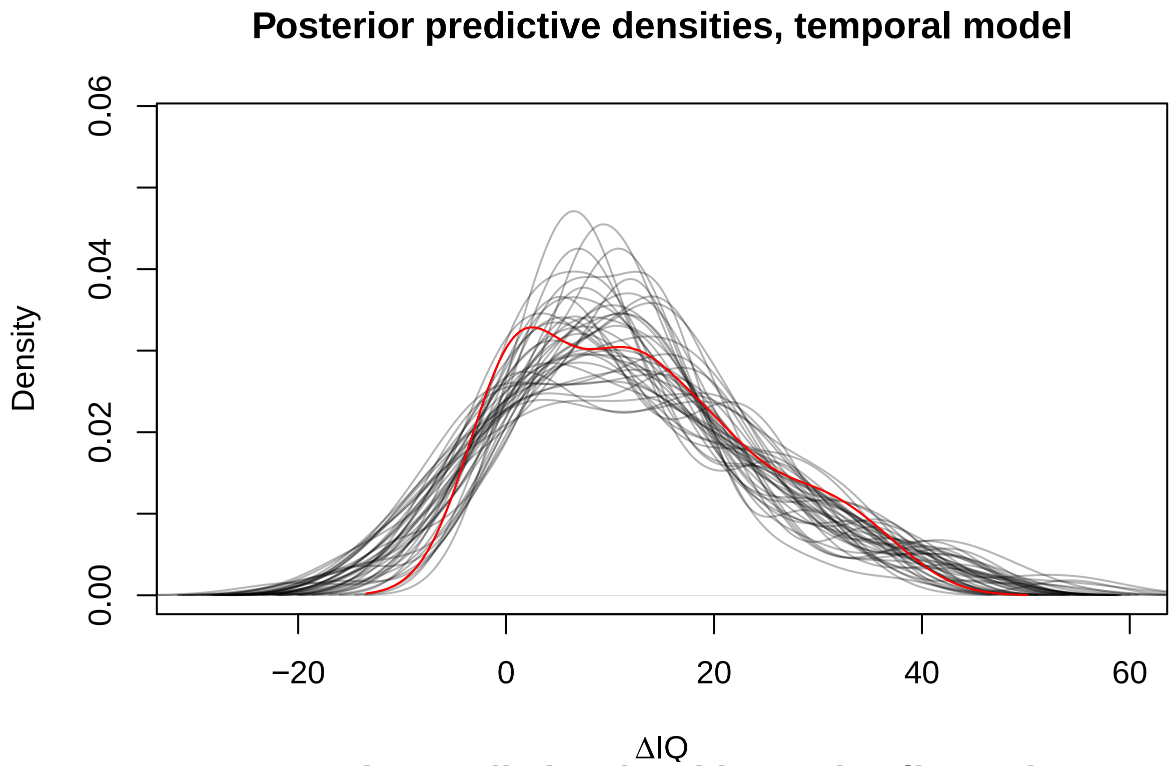


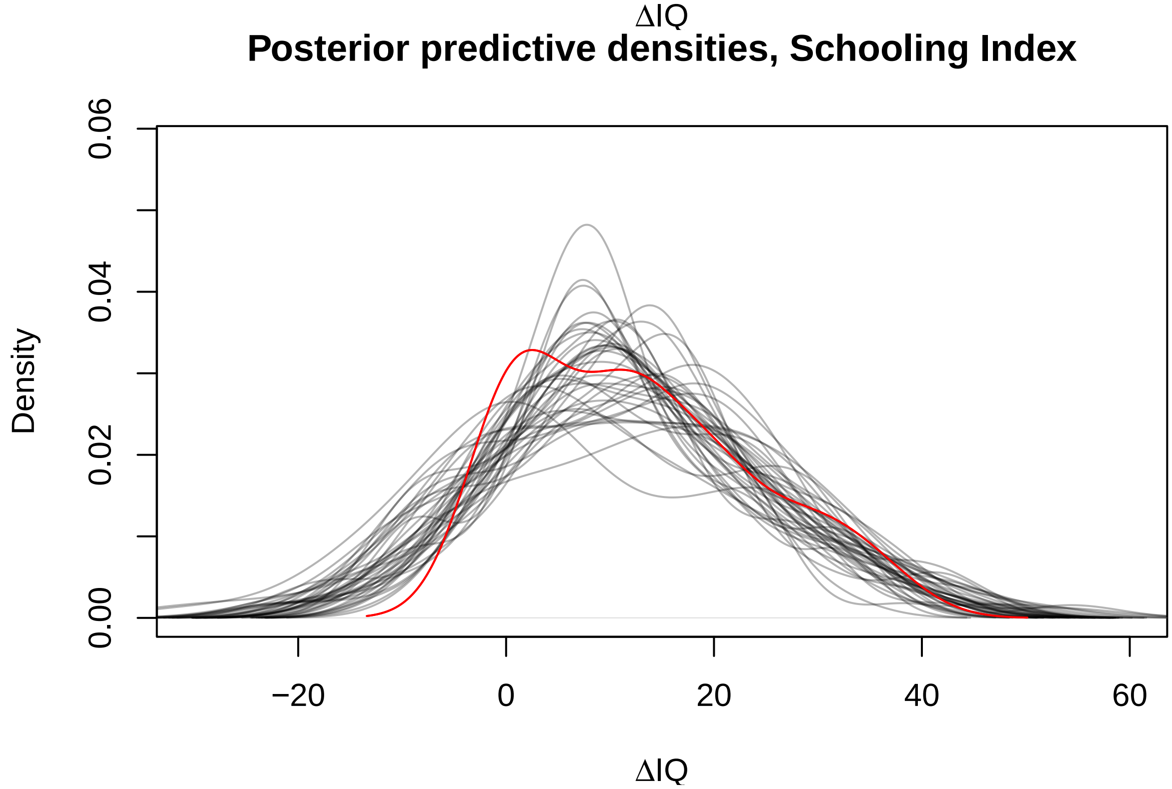
tervals.

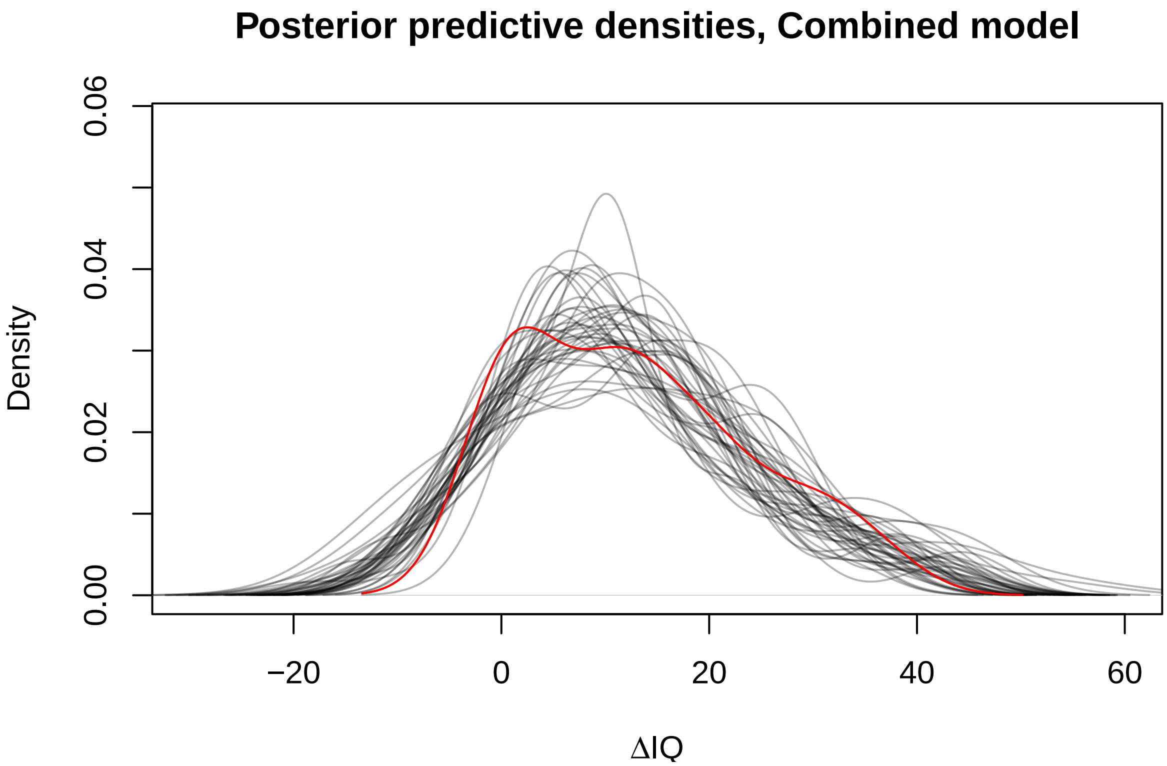
Posterior predictive checks

Visual posterior predictive checking was performed for the three pooled models due to them being the most interesting due to their best performance.

Posterior predictive checking for the Full-scale IQ Change (ΔIQ) was done by sampling 100 distributions (of 54 predicted observation) from the 4000 generated predictions. The resulting plots are below. In the plots the red line is the distribution of the original sample with the semi-transparent black lines being generated predictions.


Posterior predictive densities, temporal model
Density
0.00
0.02
0.04
0.06
−20
0
20
40
60
ΔIQ


Posterior predictive densities, Schooling Index
Density
0.00
0.02
0.04
0.06
−20
0
20
40
60
ΔIQ



The plots show that the predictions roughly follow the original distribution pretty well for the most part but with a lot of uncertainty towards the middle. These results satisfactory taking into account the limited size of the data.

Predictions

We computed predictions for IQ increases in subsequent decades for the pooled models (with only the decade index, and the model with both the decade index and the schooling index), since the pooled model infact had the best fit to the sample data. In the data, the latest observations were 10 decades from the initial starting point (decade index = 0). Therefore we predicted the IQ change 11 decades after the start point. For the simple (single predictor) pooled model, the estimate for the expected value of predicted IQ change was 37.23 while the 90% credible interval for the same was $[25.58, 48.99]$.

The prediction estimates for the other model required more involved analyses. This is because we had to come up with reliable values of the schooling index to put into the model for prediction. Since the pooled model considers all the continents interchangeable, there was no clear way to find the value of the schooling index 11 decades after the starting point. Thus we chose three sample values which were 5%, 10%, and 15% more than the latest mean value of the AHDI index. Thus we obtained three different estimates, based on the three possible (and feasible) values of the schooling index. Based on this the estimates were 36.64, 36.71, and 36.86 with 90% credible intervals $[24.6, 48.33]$, $[25.23, 48.63]$, and $[25.37, 48.25]$ respectively.

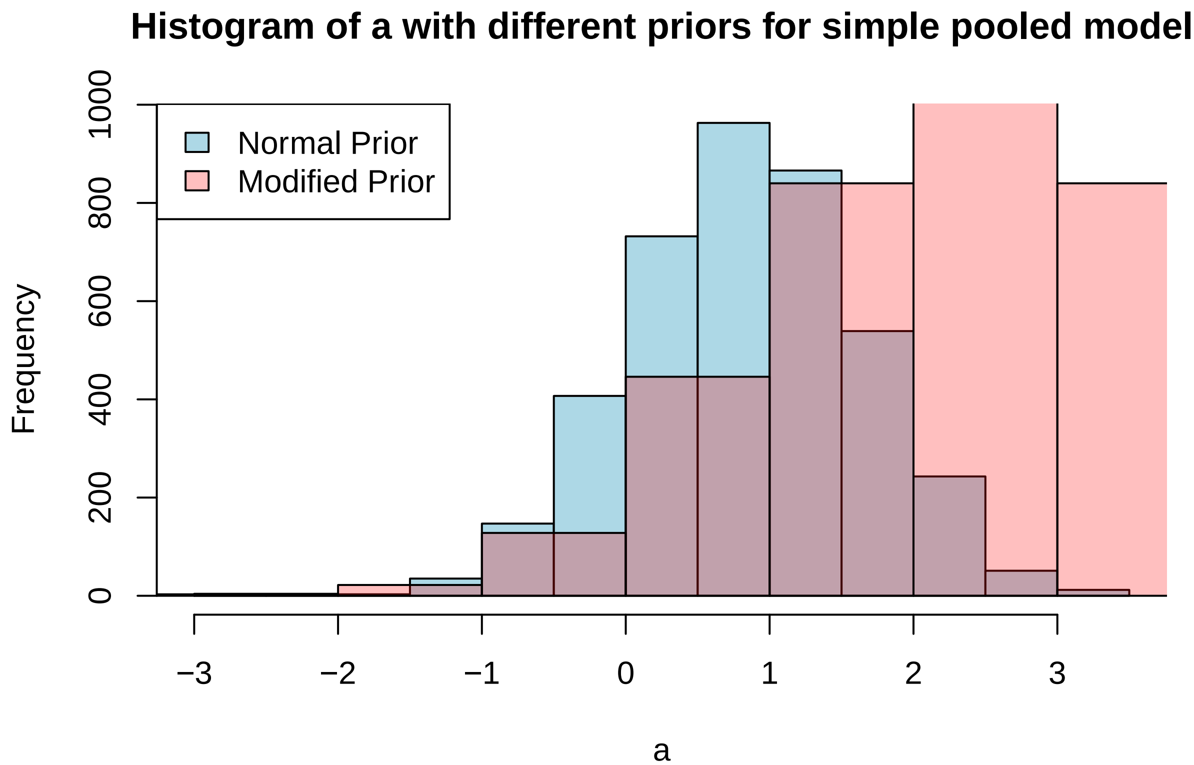
Sensitivity Analysis

Prior sensitivity analysis will be done on the simple and advanced pooled model because they are the most interesting due to their best performance. Changing priors didn't have an effect on the relative performance of the pooled, separate and hierarchical models.

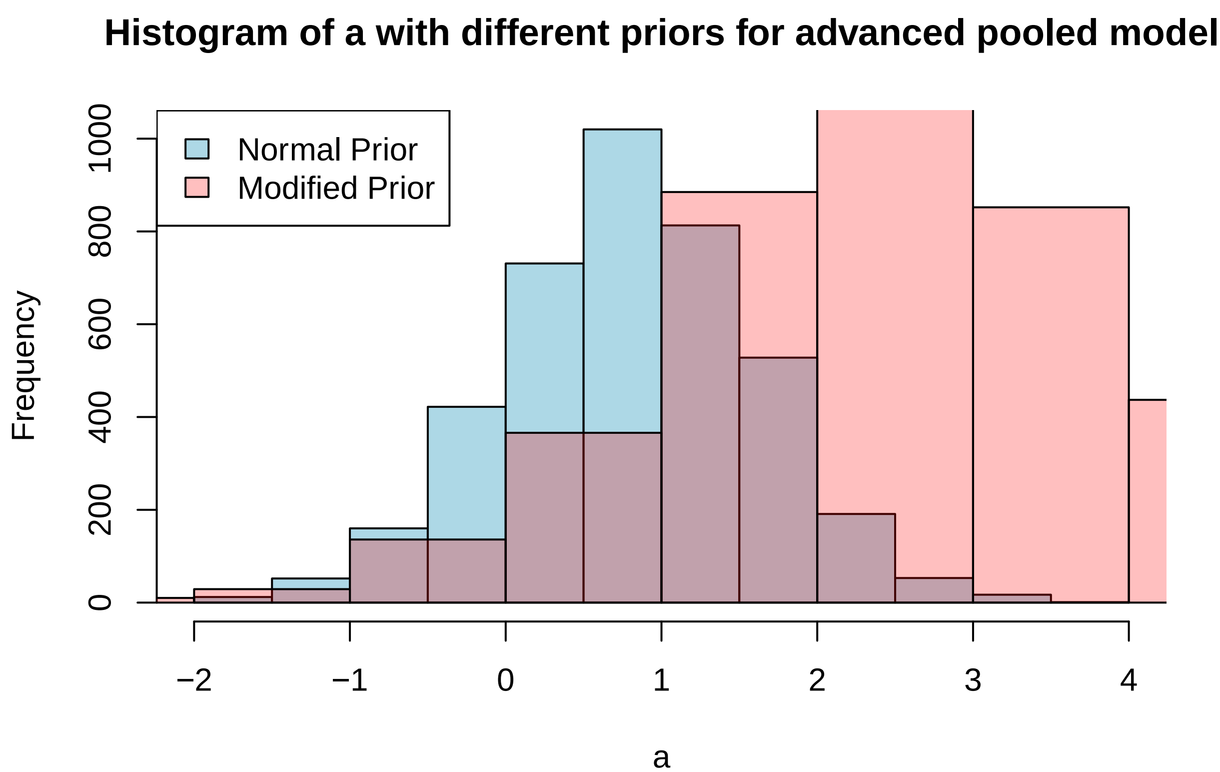
Change intercept a prior from $N(0, 1)$ to $N(0, 10)$

Reason: Even though the data starts from the 0-point, it is not necessary that the best fit line should also start there. The modified prior offers the y-intercept a bit more freedom in these regards.

Simple Model



Advanced Model

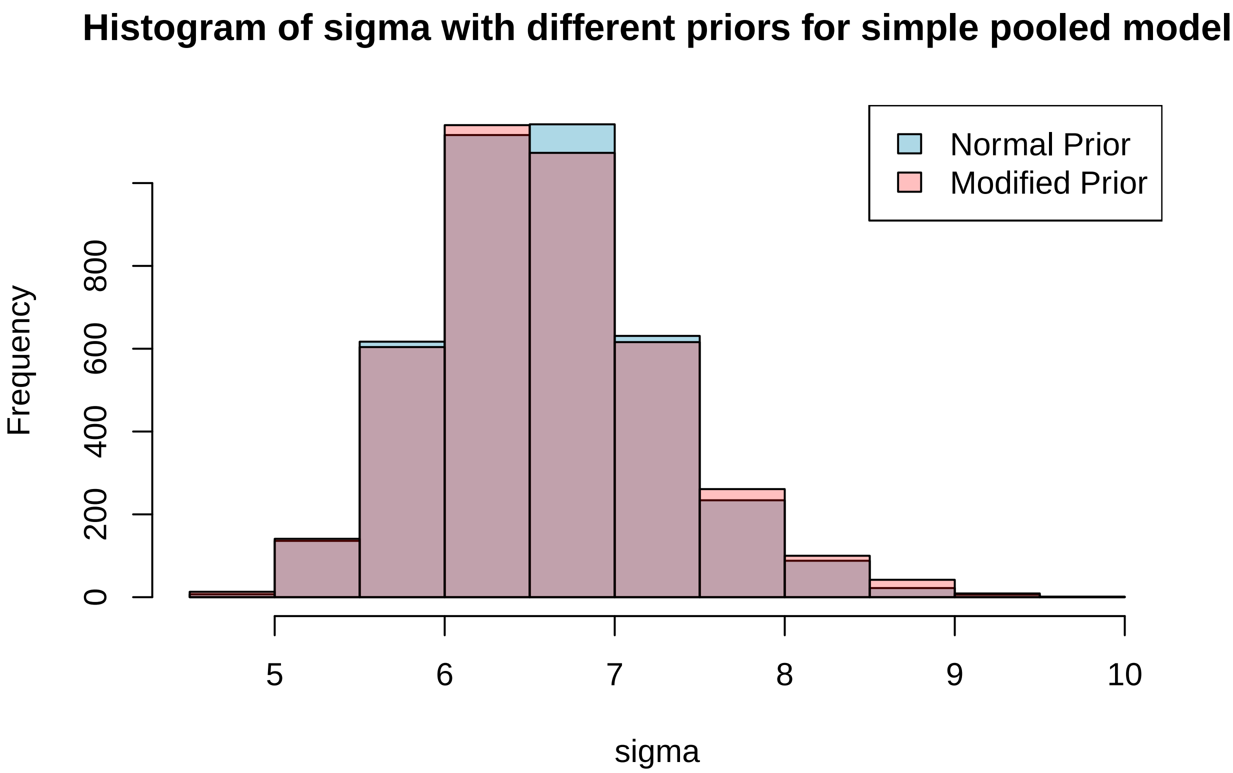


Analysis: a's distribution, when the prior is changed from $N(0,1)$ to $N(0,10)$ remains a normal distribution but shifts its center from 0.86 to 2.4. This implies that our model is sensitive to the prior distribution of a.

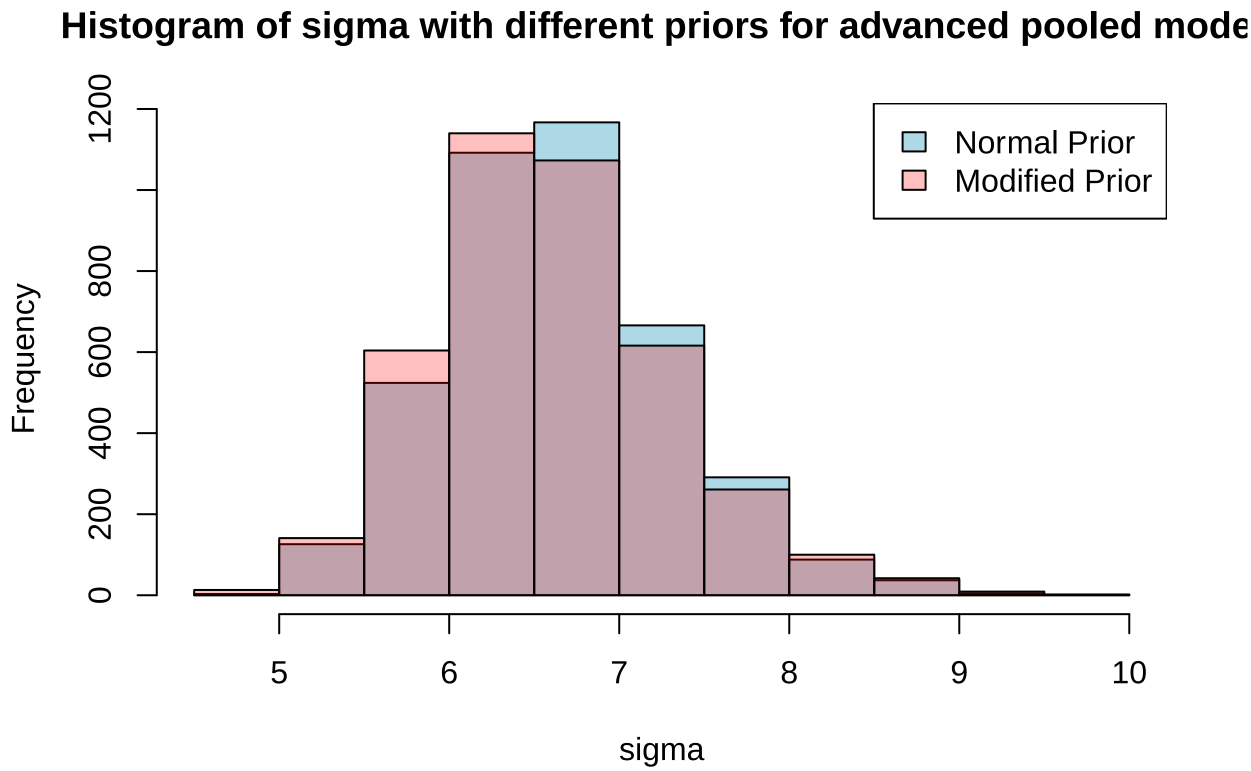
Change σ prior from $N(0,100)$ to $Inv-\chi^2(1)$

Reason: Inverse-chi squared is generally used as the prior for an unknown variance of the normal distribution. This is so because it is a conjugate prior, which for example makes the computations easier, and it also satisfies the minimal requirements needed for a prior variance.

Simple Model



Advanced Model



Analysis: σ's distribution stays almost identical when changed from $N(0, 100)$ to $inv - \chi^2(1)$. This implies that our model is insensitive to the prior distribution of sigma.

Issues and improvements

Close to the end of the project we noticed that our data had been constructed in a way that is conceptually difficult for linear regression. Mainly the same explanatory variable value would lead to different response variable values. This does not cause problems for the linear regression algorithms used but makes understanding the data more difficult. In essence a concrete uncertainty in the data. Changing this however didn't alter our results in terms of model selection or relative PSIS-LOOs so we decided to leave it as is.

Conclusion

In conclusion, we managed to find the Flynn effect in the data and find a well performing model linking schooling level to cumulative Full-scale IQ change over time. This happened to be the pooled model in our case. The pooled model performing significantly better than the other models implies that this phenomenon is global such that measurements from different continents are interchangeable. Our best performing model also predicted that the observed increase would continue with a 37.23 point increase in the next decade.

Self-reflection

We also should have consulted the TAs more but, even though we started early, we didn't have time to do this because the project started to concretize later than we expected due to our slow progress. This meant that at the point at which questions arose we no longer had time to ask them. This was due to a failure in planning and overestimating our pace. From this we learned to be better planners.

We learned that it's important to have a clear idea of what the modeling idea is and how it makes sense with the data. You can make models and run them and get results that sort of make sense without really understanding what you're doing but once you have to explain what you did you run into problems. It was an important learning experience to have to make sense of the data and model in terms people not intensely familiar with it would understand.

References

OWID. 2018. "IQ Data - Pietschnig and Voracek (2015)." https://github.com/owid/owid-datasets/tree/master/datasets/IQ%20Data%20-%20Pietschnig%20and%20Voracek%20(2015).

Pietschnig, J., and M. Voracek. 2015. "One Century of Global IQ Gains a Formal Meta-Analysis of the Flynn Effect (1909–2013)." 10: 282–306. http://pps.sagepub.com/content/10/3/282.abstract.

Prados de la Escosura, L. 2021a. "Augmented Human Development in the Age of Globalisation." *Economic History Review*. https://onlinelibrary.wiley.com/doi/full/10.1111/ehr.13064.

———. 2021b. "Dimensions of Augmented Human Development > AHDI Countries 1.1 (Xlsx)." https://frdelpino.es/investigacion/en/category/01_social-sciences/02_world-economy/03_human-development-world-economy/.

Appendix

Appendix A: Preprocessing

All data preprocessing was done in Python using standard Data Science libraries such as Numpy and Pandas with the help of SQL to combine datasets. The general workflow included first preprocessing the "Fullscale_IQ_Change" dataset, followed by preprocessing the "AHDI_countries" dataset. Next, these datasets were combined, and some minor transformations were made to make the data suit our model better. Finally it was converted to a csv file which used by all the Stan models.

Preprocessing the "Fullscale_IQ_Change" dataset was fairly easy, since it did not have any null values. Our first data transformation operation was to convert the year column to a decade column. This transformation required a trivial integer division by 10 for all the years and then multiplying them with 10. On the other hand, the "AHDI_countries" dataset required much more work. Firstly, we had to determine which continent each country belonged to, which had to be done mostly manually. Here, all the data was already listed according to decade, which reduced some preprocessing steps. Once all the countries had been classified, we replaced the countries with their continents and took the mean of the AHDI over all continents for every decade in the dataset.

Now, that both the datasets were cleaned, we needed to combine them. We determined that SQL would be the best tool for this operation. However, since we were using Python, we decided to just use the PandaSQL library which allowed us to use SQL with Pandas dataframes. This library allowed to join both the datasets with simple join query. The final change we made to the combined dataset was to change the decade column to decade index. The decade index set the first decade observation of each continent to be 0, and every subsequent decade to be incremented by 1. For example, the first decade observation for Europe was 1910 which implied the decade index 0, and the following observations from 1930, 1950, and 1960 were given decade indices 2, 4, 5 respectively. Lastly, we converted this dataframe into a csv file.

Appendix B: Stan source code

Separate temporal model:

```
data {
  int<lower=0> C; // number of continents, 5 for our data
  int<lower=0> N1; // number of observations for continent 1
  int<lower=0> N2; // number of observations for continent 2
  int<lower=0> N3; // number of observations for continent 3
  int<lower=0> N4; // number of observations for continent 4
  int<lower=0> N5; // number of observations for continent 5

  vector[N1] x1; // Decade label for continent 1
  vector[N2] x2; // Decade label for continent 2
  vector[N3] x3; // Decade label for continent 3
  vector[N4] x4; // Decade label for continent 4
  vector[N5] x5; // Decade label for continent 5

  vector[N1] y1; // IQ label for continent 1
  vector[N2] y2; // IQ label for continent 2
  vector[N3] y3; // IQ label for continent 3
  vector[N4] y4; // IQ label for continent 4
  vector[N5] y5; // IQ label for continent 5
}

parameters {
  vector[C] a; // intercepts
  vector[C] b; // slopes
  vector[C] sigma; // stds
}

transformed parameters {
  vector[N1] mu1;
  vector[N2] mu2;
  vector[N3] mu3;
  vector[N4] mu4;
  vector[N5] mu5;

  mu1 = a[1] + b[1] * x1;
  mu2 = a[2] + b[2] * x2;
  mu3 = a[3] + b[3] * x3;
  mu4 = a[4] + b[4] * x4;
  mu5 = a[5] + b[5] * x5;
}

model {
  // priors
  a ~ normal(0, 1);
  b ~ normal(0, 50);
  for (c in 1:C) {
    sigma[c] ~ normal(0, 100);
  }

  // likelihood
  y1 ~ normal(mu1, sigma[1]);
  y2 ~ normal(mu2, sigma[2]);
  y3 ~ normal(mu3, sigma[3]);
  y4 ~ normal(mu4, sigma[4]);
```

```
  y5 ~ normal(mu5, sigma[5]);
}


generated quantities {
  real y1pred[N1];
  real y2pred[N2];
  real y3pred[N3];
  real y4pred[N4];
  real y5pred[N5];
  vector[N1+N2+N3+N4+N5] log_lik;


  // posterior predictions
  y1pred = normal_rng(mu1, sigma[1]);
  y2pred = normal_rng(mu2, sigma[2]);
  y3pred = normal_rng(mu3, sigma[3]);
  y4pred = normal_rng(mu4, sigma[4]);
  y5pred = normal_rng(mu5, sigma[5]);

  // pointwise log-likelihood (log_lik)
  for (i in 1:N1) {
    log_lik[i] = normal_lpdf(y1[i] | mu1, sigma[1]);
  }

  for (i in 1:N2) {
    log_lik[i+N1] = normal_lpdf(y2[i] | mu2, sigma[2]);
  }

  for (i in 1:N3) {
    log_lik[i+N1+N2] = normal_lpdf(y3[i] | mu3, sigma[3]);
  }

  for (i in 1:N4) {
    log_lik[i+N1+N2+N3] = normal_lpdf(y4[i] | mu4, sigma[4]);
  }
  for (i in 1:N5) {
    log_lik[i+N1+N2+N3+N4] = normal_lpdf(y5[i] | mu5, sigma[5]);
  }
}
```

Separate model with schooling index:

```
data {
  int<lower=0> C; // number of continents, 5 for our data
  int<lower=0> N1; // number of observations for continent 1
  int<lower=0> N2; // number of observations for continent 2
  int<lower=0> N3; // number of observations for continent 3
  int<lower=0> N4; // number of observations for continent 4
  int<lower=0> N5; // number of observations for continent 5

  vector[N1] y1; // IQ label for continent 1
  vector[N2] y2; // IQ label for continent 2
  vector[N3] y3; // IQ label for continent 3
```

```
  vector[N4] y4; // IQ label for continent 4
  vector[N5] y5; // IQ label for continent 5

  vector[N1] z1; // Schooling level label for continent 1
  vector[N2] z2; // Schooling level for continent 2
  vector[N3] z3; // Schooling level for continent 3
  vector[N4] z4; // Schooling level for continent 4
  vector[N5] z5; // Schooling level for continent 5
}

parameters {
  vector[C] a; // intercepts
  vector[C] c; // schooling level slopes

  vector<lower=0>[C] sigma; // stds
}

transformed parameters {
  vector[N1] mu1;
  vector[N2] mu2;
  vector[N3] mu3;
  vector[N4] mu4;
  vector[N5] mu5;

  mu1 = a[1] + c[1] * z1;
  mu2 = a[2] + c[2] * z2;
  mu3 = a[3] + c[3] * z3;
  mu4 = a[4] + c[4] * z4;
  mu5 = a[5] + c[5] * z5;
}

model {
  // priors
  a ~ normal(0, 1);
  c ~ normal(0, 100);

  // mu ~ normal(0, 10);
  //for (i in 1:C) {
  // this is informative?  sigma[i] ~ inv_chi_square(1);
  //}
  for (i in 1:C) {
   sigma[i] ~ normal(0, 100);
  }

  // likelihood
  y1 ~ normal(mu1, sigma[1]);
  y2 ~ normal(mu2, sigma[2]);
  y3 ~ normal(mu3, sigma[3]);
  y4 ~ normal(mu4, sigma[4]);
  y5 ~ normal(mu5, sigma[5]);
}


generated quantities {
```

```
  real y1pred[N1];
  real y2pred[N2];
  real y3pred[N3];
  real y4pred[N4];
  real y5pred[N5];
  vector[N1+N2+N3+N4+N5] log_lik;

  // posterior predictions
  y1pred = normal_rng(mu1, sigma[1]);
  y2pred = normal_rng(mu2, sigma[2]);
  y3pred = normal_rng(mu3, sigma[3]);
  y4pred = normal_rng(mu4, sigma[4]);
  y5pred = normal_rng(mu5, sigma[5]);
  // pointwise log-likelihood (log_lik)
  for (i in 1:N1) {
    log_lik[i] = normal_lpdf(y1[i] | mu1, sigma[1]);
  }

  for (i in 1:N2) {
    log_lik[i+N1] = normal_lpdf(y2[i] | mu2, sigma[2]);
  }

  for (i in 1:N3) {
    log_lik[i+N1+N2] = normal_lpdf(y3[i] | mu3, sigma[3]);
  }

  for (i in 1:N4) {
    log_lik[i+N1+N2+N3] = normal_lpdf(y4[i] | mu4, sigma[4]);
  }
  for (i in 1:N5) {
    log_lik[i+N1+N2+N3+N4] = normal_lpdf(y5[i] | mu5, sigma[5]);
  }
}
```

Separate combined model:

```
data {
  int<lower=0> C; // number of continents, 5 for our data
  int<lower=0> N1; // number of observations for continent 1
  int<lower=0> N2; // number of observations for continent 2
  int<lower=0> N3; // number of observations for continent 3
  int<lower=0> N4; // number of observations for continent 4
  int<lower=0> N5; // number of observations for continent 5

  vector[N1] x1; // Decade label for continent 1
  vector[N2] x2; // Decade label for continent 2
  vector[N3] x3; // Decade label for continent 3
  vector[N4] x4; // Decade label for continent 4
  vector[N5] x5; // Decade label for continent 5

  vector[N1] y1; // IQ label for continent 1
  vector[N2] y2; // IQ label for continent 2
  vector[N3] y3; // IQ label for continent 3
  vector[N4] y4; // IQ label for continent 4
```

```
  vector[N5] y5; // IQ label for continent 5

  vector[N1] z1; // Schooling level label for continent 1
  vector[N2] z2; // Schooling level for continent 2
  vector[N3] z3; // Schooling level for continent 3
  vector[N4] z4; // Schooling level for continent 4
  vector[N5] z5; // Schooling level for continent 5
}

parameters {
  vector[C] a; // intercepts
  vector[C] b; // decade slopes
  vector[C] c; // schooling level slopes

  vector<lower=0>[C] sigma; // stds
}

transformed parameters {
  vector[N1] mu1;
  vector[N2] mu2;
  vector[N3] mu3;
  vector[N4] mu4;
  vector[N5] mu5;

  mu1 = a[1] + b[1] * x1 + c[1] * z1;
  mu2 = a[2] + b[2] * x2 + c[2] * z2;
  mu3 = a[3] + b[3] * x3 + c[3] * z3;
  mu4 = a[4] + b[4] * x4 + c[4] * z4;
  mu5 = a[5] + b[5] * x5 + c[5] * z5;
}

model {
  // priors
  a ~ normal(0, 1);
  b ~ normal(0, 50);
  c ~ normal(0, 100);

  // mu ~ normal(0, 10);
  //for (i in 1:C) {
  // this is informative?  sigma[i] ~ inv_chi_square(1);
  //}
  for (i in 1:C) {
    sigma[i] ~ normal(0, 100);
  }

  // likelihood
  y1 ~ normal(mu1, sigma[1]);
  y2 ~ normal(mu2, sigma[2]);
  y3 ~ normal(mu3, sigma[3]);
  y4 ~ normal(mu4, sigma[4]);
  y5 ~ normal(mu5, sigma[5]);
}

generated quantities {
```

```
  real y1pred[N1];
  real y2pred[N2];
  real y3pred[N3];
  real y4pred[N4];
  real y5pred[N5];
  vector[N1+N2+N3+N4+N5] log_lik;

  // posterior predictions
  y1pred = normal_rng(mu1, sigma[1]);
  y2pred = normal_rng(mu2, sigma[2]);
  y3pred = normal_rng(mu3, sigma[3]);
  y4pred = normal_rng(mu4, sigma[4]);
  y5pred = normal_rng(mu5, sigma[5]);
  // pointwise log-likelihood (log_lik)
    for (i in 1:N1) {
    log_lik[i] = normal_lpdf(y1[i] | mu1, sigma[1]);
  }

  for (i in 1:N2) {
    log_lik[i+N1] = normal_lpdf(y2[i] | mu2, sigma[2]);
  }

  for (i in 1:N3) {
    log_lik[i+N1+N2] = normal_lpdf(y3[i] | mu3, sigma[3]);
  }

  for (i in 1:N4) {
    log_lik[i+N1+N2+N3] = normal_lpdf(y4[i] | mu4, sigma[4]);
  }
  for (i in 1:N5) {
    log_lik[i+N1+N2+N3+N4] = normal_lpdf(y5[i] | mu5, sigma[5]);
  }
}
```

Hierarchical temporal model:

```
data {
  // int<lower=0> N; // Number of observations per continent (different for each one!)
  int<lower=0> C; // Number of continents

  int<lower=0> N1; // number of observations for continent 1
  int<lower=0> N2; // number of observations for continent 2
  int<lower=0> N3; // number of observations for continent 3
  int<lower=0> N4; // number of observations for continent 4
  int<lower=0> N5; // number of observations for continent 5

  vector[N1] x1; // Decade label for continent 1
  vector[N2] x2; // Decade label for continent 2
  vector[N3] x3; // Decade label for continent 3
  vector[N4] x4; // Decade label for continent 4
  vector[N5] x5; // Decade label for continent 5

  vector[N1] y1; // IQ label for continent 1
  vector[N2] y2; // IQ label for continent 2
```

```
  vector[N3] y3; // IQ label for continent 3
  vector[N4] y4; // IQ label for continent 4
  vector[N5] y5; // IQ label for continent 5

}

parameters {
  real<lower=0> sigma;
  vector[C] a; // Intercept
  vector[C] b; // Decade slope

  vector[N1] mu1;
  vector[N2] mu2;
  vector[N3] mu3;
  vector[N4] mu4;
  vector[N5] mu5;
}

model {
  // Priors
  a ~ normal(0, 1);
  b ~ normal(0, 50);

  sigma ~ normal(0, 100); // Shared variance

  mu1 ~ normal(a[1] + b[1] * x1, sigma);
  mu2 ~ normal(a[2] + b[2] * x2, sigma);
  mu3 ~ normal(a[3] + b[3] * x3, sigma);
  mu4 ~ normal(a[4] + b[4] * x4, sigma);
  mu5 ~ normal(a[5] + b[5] * x5, sigma);
  // likelihood
  y1 ~ normal(mu1, 1);
  y2 ~ normal(mu2, 1);
  y3 ~ normal(mu3, 1);
  y4 ~ normal(mu4, 1);
  y5 ~ normal(mu5, 1);
}

generated quantities {
  vector[N1+N2+N3+N4+N5] log_lik;

  for (i in 1:N1) {
    log_lik[i] = normal_lpdf(y1[i] | mu1, 1);
  }

  for (i in 1:N2) {
    log_lik[i+N1] = normal_lpdf(y2[i] | mu2, 1);
  }

  for (i in 1:N3) {
    log_lik[i+N1+N2] = normal_lpdf(y3[i] | mu3, 1);
  }

  for (i in 1:N4) {
```

```
    log_lik[i+N1+N2+N3] = normal_lpdf(y4[i] | mu4, 1);
  }
  for (i in 1:N5) {
    log_lik[i+N1+N2+N3+N4] = normal_lpdf(y5[i] | mu5, 1);
  }
}
```

Hierarchical model with schooling index:

```
data {
  // int<lower=0> N; // Number of observations per continent (different for each one!)
  int<lower=0> C; // Number of continents

  int<lower=0> N1; // number of observations for continent 1
  int<lower=0> N2; // number of observations for continent 2
  int<lower=0> N3; // number of observations for continent 3
  int<lower=0> N4; // number of observations for continent 4
  int<lower=0> N5; // number of observations for continent 5

  vector[N1] y1; // IQ label for continent 1
  vector[N2] y2; // IQ label for continent 2
  vector[N3] y3; // IQ label for continent 3
  vector[N4] y4; // IQ label for continent 4
  vector[N5] y5; // IQ label for continent 5

  vector[N1] z1; // Schooling level label for continent 1
  vector[N2] z2; // Schooling level for continent 2
  vector[N3] z3; // Schooling level for continent 3
  vector[N4] z4; // Schooling level for continent 4
  vector[N5] z5; // Schooling level for continent 5
}

parameters {
  real<lower=0> sigma;
  vector[C] a; // Intercept
  vector[C] c; // Schooling level slope

  vector[N1] mu1;
  vector[N2] mu2;
  vector[N3] mu3;
  vector[N4] mu4;
  vector[N5] mu5;
}

model {
  // Priors
  a ~ normal(0, 1);
  c ~ normal(0, 100);

  sigma ~ normal(0, 100); // Shared variance

  mu1 ~ normal(a[1] + c[1] * z1, sigma);
  mu2 ~ normal(a[2] + c[2] * z2, sigma);
  mu3 ~ normal(a[3] + c[3] * z3, sigma);
```

```
  mu4 ~ normal(a[4] + c[4] * z4, sigma);
  mu5 ~ normal(a[5] + c[5] * z5, sigma);
  // likelihood
  y1 ~ normal(mu1, 1);
  y2 ~ normal(mu2, 1);
  y3 ~ normal(mu3, 1);
  y4 ~ normal(mu4, 1);
  y5 ~ normal(mu5, 1);
}
generated quantities {
  vector[N1+N2+N3+N4+N5] log_lik;

  for (i in 1:N1) {
    log_lik[i] = normal_lpdf(y1[i] | mu1, 1);
  }

  for (i in 1:N2) {
    log_lik[i+N1] = normal_lpdf(y2[i] | mu2, 1);
  }

  for (i in 1:N3) {
    log_lik[i+N1+N2] = normal_lpdf(y3[i] | mu3, 1);
  }

  for (i in 1:N4) {
    log_lik[i+N1+N2+N3] = normal_lpdf(y4[i] | mu4, 1);
  }
  for (i in 1:N5) {
    log_lik[i+N1+N2+N3+N4] = normal_lpdf(y5[i] | mu5, 1);
  }
}
```

Hierarchical combined model:

```
data {
  // int<lower=0> N; // Number of observations per continent (different for each one!)
  int<lower=0> C; // Number of continents

  int<lower=0> N1; // number of observations for continent 1
  int<lower=0> N2; // number of observations for continent 2
  int<lower=0> N3; // number of observations for continent 3
  int<lower=0> N4; // number of observations for continent 4
  int<lower=0> N5; // number of observations for continent 5

  vector[N1] x1; // Decade label for continent 1
  vector[N2] x2; // Decade label for continent 2
  vector[N3] x3; // Decade label for continent 3
  vector[N4] x4; // Decade label for continent 4
  vector[N5] x5; // Decade label for continent 5

  vector[N1] y1; // IQ label for continent 1
  vector[N2] y2; // IQ label for continent 2
  vector[N3] y3; // IQ label for continent 3
  vector[N4] y4; // IQ label for continent 4
```

```
  vector[N5] y5; // IQ label for continent 5

  vector[N1] z1; // Schooling level label for continent 1
  vector[N2] z2; // Schooling level for continent 2
  vector[N3] z3; // Schooling level for continent 3
  vector[N4] z4; // Schooling level for continent 4
  vector[N5] z5; // Schooling level for continent 5
}

parameters {
  real<lower=0> sigma;
  vector[C] a; // Intercept
  vector[C] b; // Decade slope
  vector[C] c; // Schooling level slope

  vector[N1] mu1;
  vector[N2] mu2;
  vector[N3] mu3;
  vector[N4] mu4;
  vector[N5] mu5;
}

model {
  // Priors
  a ~ normal(0, 1);
  b ~ normal(0, 50);
  c ~ normal(0, 100);

  sigma ~ normal(0, 100); // Shared variance

  mu1 ~ normal(a[1] + b[1] * x1 + c[1] * z1, sigma);
  mu2 ~ normal(a[2] + b[2] * x2 + c[2] * z2, sigma);
  mu3 ~ normal(a[3] + b[3] * x3 + c[3] * z3, sigma);
  mu4 ~ normal(a[4] + b[4] * x4 + c[4] * z4, sigma);
  mu5 ~ normal(a[5] + b[5] * x5 + c[5] * z5, sigma);
  // likelihood
  y1 ~ normal(mu1, 1);
  y2 ~ normal(mu2, 1);
  y3 ~ normal(mu3, 1);
  y4 ~ normal(mu4, 1);
  y5 ~ normal(mu5, 1);
}
generated quantities {
  vector[N1+N2+N3+N4+N5] log_lik;

  for (i in 1:N1) {
    log_lik[i] = normal_lpdf(y1[i] | mu1, 1);
  }

  for (i in 1:N2) {
    log_lik[i+N1] = normal_lpdf(y2[i] | mu2, 1);
  }

  for (i in 1:N3) {
```

```
    log_lik[i+N1+N2] = normal_lpdf(y3[i] | mu3, 1);
  }

  for (i in 1:N4) {
    log_lik[i+N1+N2+N3] = normal_lpdf(y4[i] | mu4, 1);
  }
  for (i in 1:N5) {
    log_lik[i+N1+N2+N3+N4] = normal_lpdf(y5[i] | mu5, 1);
  }
}
```

Pooled temporal model:

```
data{
  int<lower=0> N;  // no of observations * no of observations
  vector[N] x1;     // Decade label
  vector[N] y;     // IQ label
  int xpred;       // Decade for prediction
}

parameters{
  real a;              // intercept
  real b;              // slope of decade
  real<lower=0> sigma; // std
}

transformed parameters{
  vector[N] mu;
  mu = a + b*x1;
}

model{
  a ~ normal(0, 1);
  b ~ normal(0, 50);

  sigma ~ normal(0,100);

  // likelihood
  y ~ normal(mu, sigma);

}

generated quantities {
  real ypred;
  vector [N] log_lik;
  vector [N] y_pred;

  for (i in 1:N){
    log_lik[i] = normal_lpdf(y[i] | mu[i], sigma);
    y_pred[i] = normal_rng(mu[i], sigma);
  }
  ypred = normal_rng(a+b*xpred, sigma);

}
```

Pooled model with schooling index:

```
data{
  int<lower=0> N;  // no of observations * no of observations
  vector[N] x1;     // Schooling Index label
  vector[N] y;     // IQ label
}

parameters{
  real a;              // intercept
  real c;              // slope of schooling index
  real<lower=0> sigma; // std
}

transformed parameters{
  vector[N] mu;
  mu = a + c*x1;
}

model{
  a ~ normal(0, 1);
  c ~ normal(0, 100);

  sigma ~ normal(0,100);

  // likelihood
  y ~ normal(mu, sigma);

}

generated quantities {
  vector [N] y_pred;
  vector [N] log_lik;

  for (i in 1:N){
    y_pred[i] = normal_rng(mu[i], sigma);
    log_lik[i] = normal_lpdf(y[i] | mu[i], sigma);
  }
}
```

Pooled combined model:

```
data{
  int<lower=0> N;  // no of continents * no of observations
  vector[N] x1;    // Decade label
  vector[N] x2;    // Schooling Index
  vector[N] y;     // IQ label
  real xpred_decade; // Prediction Decade
  real xpred_schooling [3]; // Schooling index with 5/10/15% increase
}

parameters{
  real a;              // intercept
```

```
  real b;              // slope of decade
  real c;              // slope of schooling index
  real<lower=0> sigma; // std
}

transformed parameters{
  vector[N] mu;
  mu = a + b*x1 + c*x2;
}

model{
  a ~ normal(0, 1);
  b ~ normal(0, 50);
  c ~ normal(0, 100);

  sigma ~ normal(0,100);

  // likelihood
  y ~ normal(mu, sigma);

}

generated quantities {
  real ypred[3];
  vector[N] y_pred;
  vector [N] log_lik;

  for (i in 1:N){
    y_pred[i] = normal_rng(mu[i], sigma);
    log_lik[i] = normal_lpdf(y[i] | mu[i], sigma);
  }
  for (i in 1:3){
   ypred[i] = normal_rng(a+b*xpred_decade + c*xpred_schooling[i], sigma);
  }
}
```