Predicting Weight to Height Z-Score in the Global South

Sam Loescher and Camiel Schroeder

Introduction

Humans have struggled with malnutrition since before the dawn of civilization. The root causes of the problem have shifted, however. The issue is no longer our limited ability to grow food. Humanity now produces more than enough food to adequately feed every living human, yet the problem of malnutrition persists. Rather than true scarcity, economic inequality and poor redistribution of resources are now the key drivers of malnutrition. One way to remedy this resource allocation inequality is through direct cash transfers (DCT). Researchers in Jharkhand, India researched the effects of DCTs to new parents on numerous health outcomes, including the weight to height z-score (WHZ) of children.

The effects of DCTs, while important, are well documented. Giving cash to people in poverty improves their health outcomes. The researchers' data set is extensive, however, which opens the possibility to further statistical analysis beyond the temporal effects of DCTs.

After exploring many variables, we chose mother's education level (measured in years in school), happiness level (a self-prescribed score from 1-5), and ration card status (an effective measure of overall socioeconomic standing) as our explanatory variables because they represented key indicators of different aspect of life which logically seem like they could explain large parts of why some children are undernourished while others are not. We chose WHZ as the response variable because it is a well-validated indicator of nutritional status among children, especially very small ones.

Thus, rather than observing change over time, we used the data to attempt to predict a child's HWZ given their family's ration card status (A marker of socioeconomic standing), their mother's education level, and her happiness level. If we were able to predict these with low error, this could indicate that the global health community should target these variables when seeking to improve young children's WHZ in poor communities.

Statistical procedures used

In this study, the response variable is the height to weight z-score of children in Jharkhand, India, and the explanatory variables are mother's education level, father's education level, empowerment index, and distance from market. These variables were all measured on the 3,142 families in our chosen subset of the study. These families represent the observational unit in this study. To address the research question, we implemented a K-nearest neighbors (KNN) test.

The KNN algorithm calculates how different all of the points in the data set are from one unknown point, and then takes the mean of some number (K) of the most similar points. For instance, if we wanted to predict the WHZ for a child with AAY ratio status (the poorest classification) whose mother had 10 years of education, and rated her happiness at two out of five, we would find the K most similar ("nearest") points, and take their mean WHZ to create our prediction.

One key assumption in any KNN algorithm is that there is enough data for the number of dimensions (explanatory variables). If there are too many dimensions for the amount of data provided, then there is a chance that there will not be many near neighbors to the target point. This will make the predictions much less accurate, and often necessitates removing dimensions from the analysis. Our chosen variables do not suffer from this problem because they are non-continuous with relatively low scales. Happiness score ranges from 1-5, education level from 0-16, and there are three ration card statuses. Thus, many combinations of our explanatory variables actually have many observations. This is a key determinant of our optimal K value, and is discussed further in our summary of statistical findings.

There are many ways of calculating how similar points are to one another (the "distance"). For this report, we used the Gower Distance. Gower distance standardizes the variable scale among numeric variables, and is able to account for categorical variables in its distance measurement. This makes it perfect for the complexity of data set we are working with, with many types of data and different scales involved.

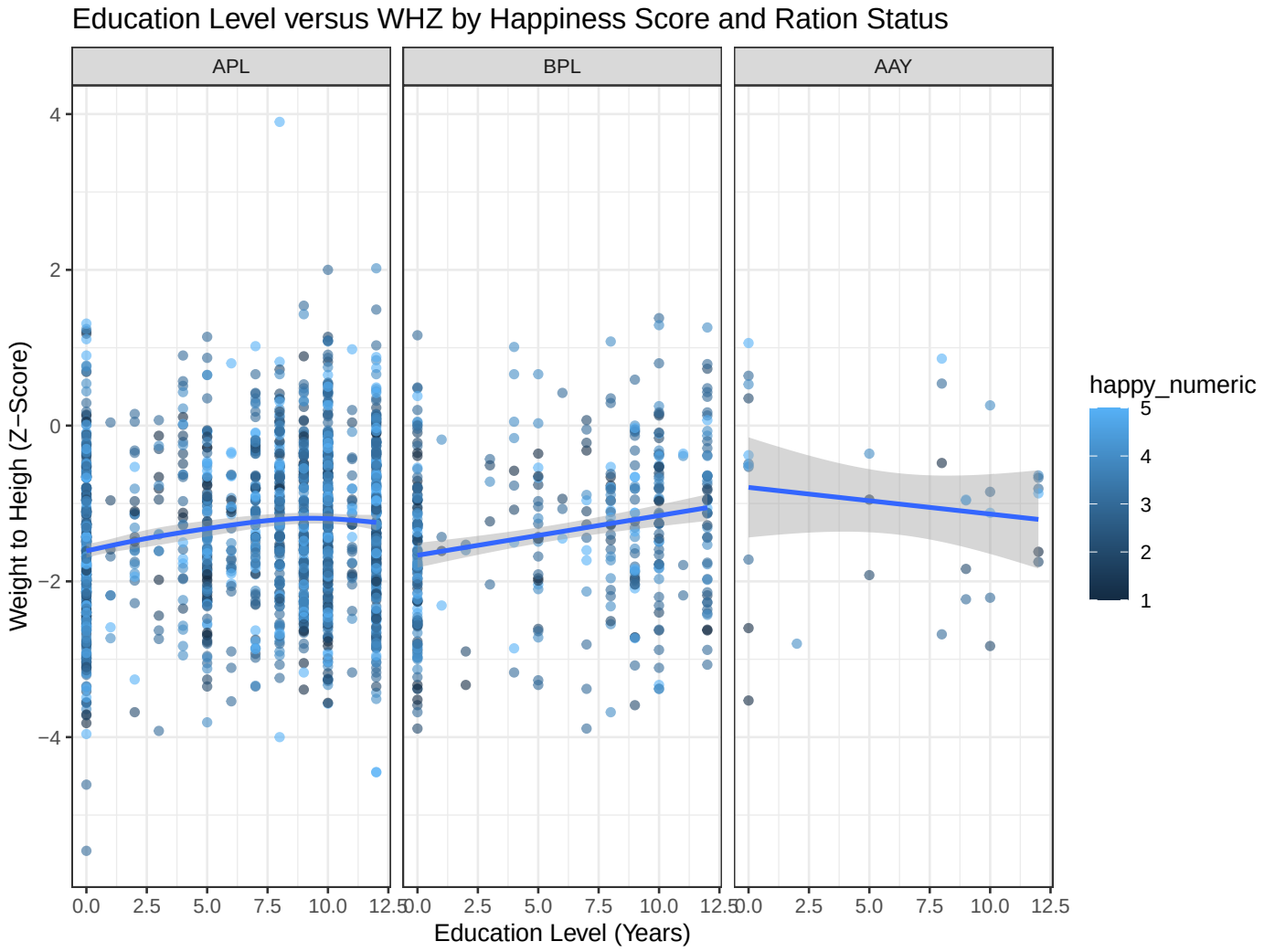


Figure caption.

This graph shows all of the variables included in our analysis. It indicated a moderate increase in the WHZ of children as their mother's education level increases for the two poorest ration status groups. The third ration status group is more wealthy and has fewer data points, so the apparent negative

relationship does not disprove the overall positive relationship.

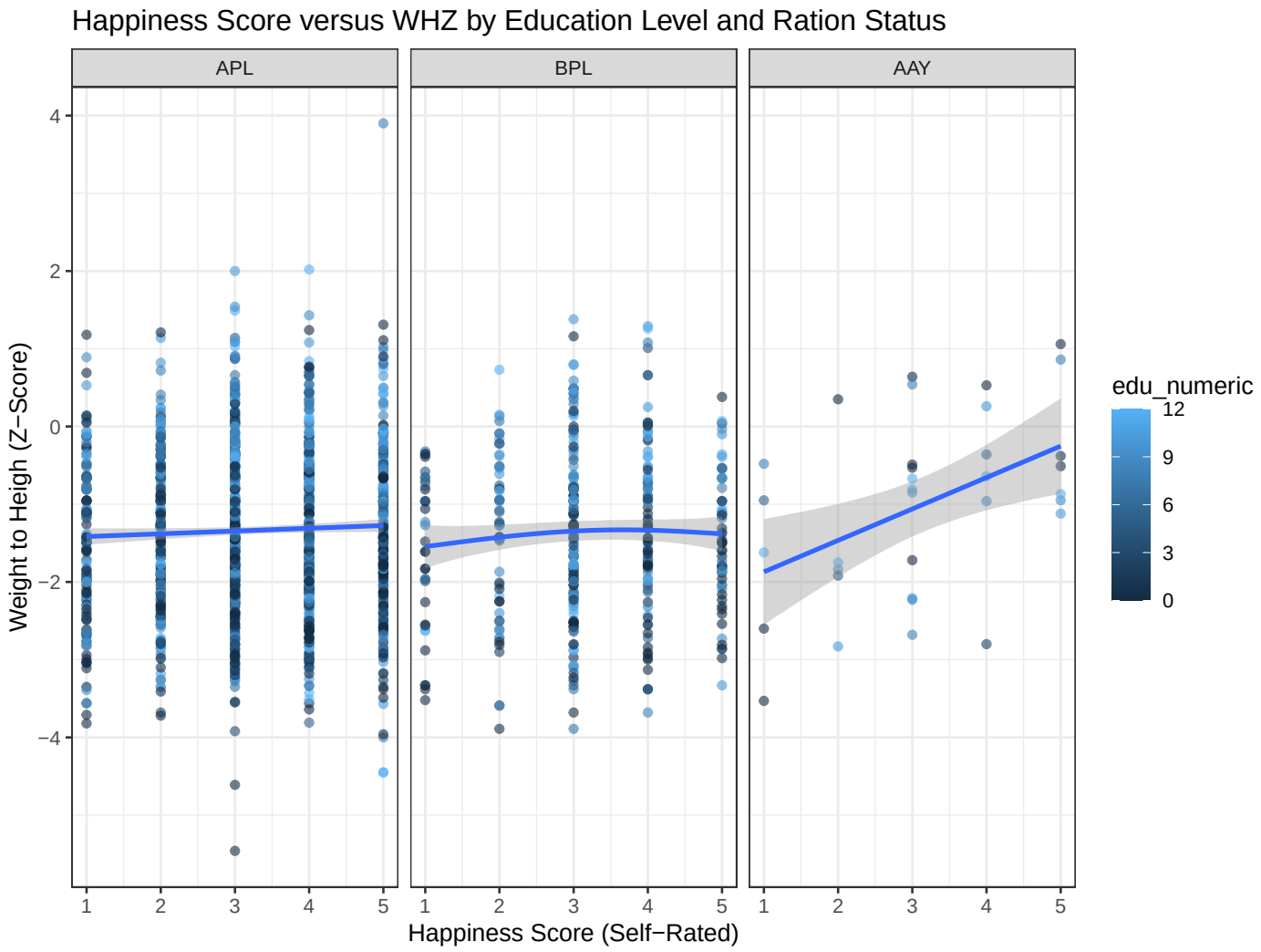
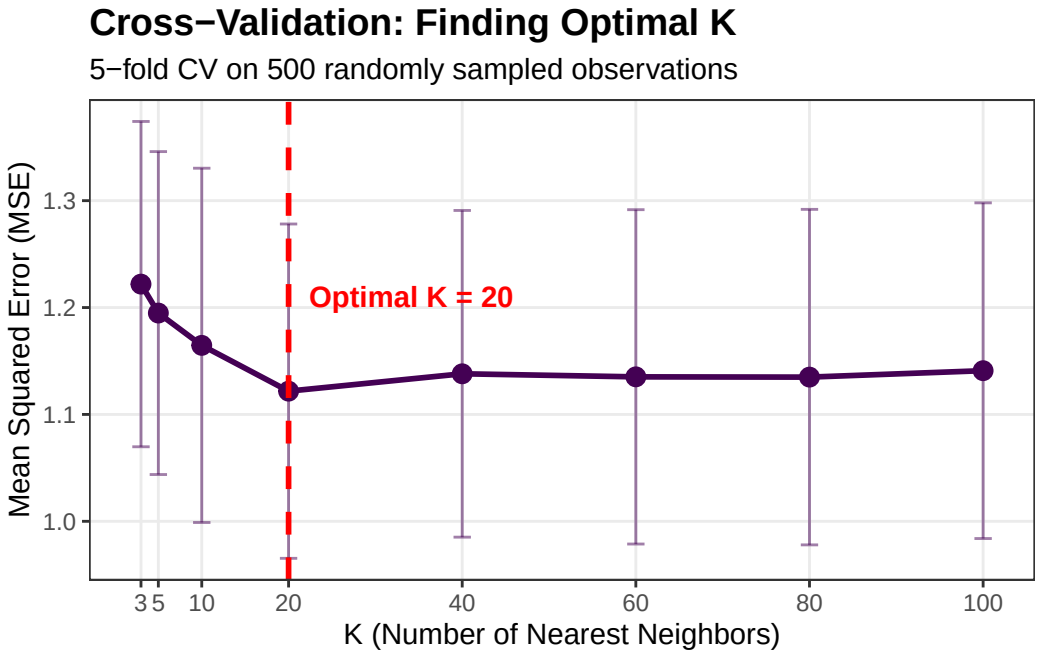


Figure caption.

This graph shows the relationship between Happiness Score and WHZ more clearly, with education level displayed in colors. This graph shows an apparent very small but positive relationship between Happiness Score and WHZ across the ration status groups. The most wealthy ration status group again displays a distinct trend from the other two, this time with a much more positive slope. This apparent relationship necessitates further study beyond the scope of this paper.

In order to determine the K-value for our algorithm, we calculated the mean squared error (MSE) for 9 k-values ranging from 3 to 100 using a subset of our data set.

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```



The prediction error (MSE) decreases as K increases, but this decrease leveles out from K=20-100. This makes sense given the many data points with identical values for all of our explanatory variables. If there are more neighbor points with a distance of zero than the K-value selected, then changes in the K-Value within that range would not change the result.

K=20 was selected as optimal, achieving an MSE of 1.03 while maintaining local prediction characteristics. This represents using approximately 5% of the training data for each prediction, balancing bias-variance tradeoff effectively. Our method for calculating the error for each K is available in the appendix.

Summary of Statistical Findings

The KNN algorithm's optimal MSE of 1.03 is almost equal to a naive estimator of WHZ: the variance. This means that using these three predictors to find points similar to the target point does no better

than simply predicting the mean value for every data point, regardless of the participant's education level, happiness index, or ration card status. Thus we conclude that the KNN algorithm using these predictors should not be used to make predictions about the WHZ of children, nor should any correlation between the variables and malnutrition be accepted on the basis of this analysis.

Discussion

We attempted to create a KNN algorithm that predicted a child's weight to height z-score based on their household's ration-card status and their mother's education level and happiness score. While we successfully created such an algorithm and tuned it for the best K value, it failed to outperform a naive estimator. This does not mean that these variables, or others like them, couldn't be used to predict WHZ. Further research should test more variables to determine the strongest correlations, and should investigate other modes of prediction besides KNN. Linear regression and other non-linear approaches should be explored.

Acknowledgements and Author Contributions

Sam and Camiel worked on this report together. Sam capitalized on the rain trapping Professor Stratton in Monroe to have a very length discussion about the KNN algorithm in office hours. Sam and Camiel both used ChatGPT for debugging, and for conceptual understanding of in-class coding examples.

Appendix

Code

```
data_old <- read_dta("p1y2_endline_field_hh_PUBLIC.dta")
data_2 <- data_old%>%
  mutate(happy_numeric = as.numeric(happy_likert)) %>%
  mutate(edu_numeric = as.numeric(resp_edu)) %>%
  dplyr::select(uid, happy_numeric, edu_numeric) %>%
  filter(!is.na(happy_numeric))%>%
  filter(!is.na(edu_numeric))


p1y1_end_anth_old <- read_dta("p1y1_endline_anthropometrics.dta")
p1y1_end_anth <- p1y1_end_anth_old%>%mutate(whz_numeric = as.numeric(whz)) %>%
  dplyr::select(uid, whz_numeric) %>%
  filter(!is.na(whz_numeric))%>%
  distinct(uid, .keep_all = TRUE)


  data <- left_join(data_2, p1y1_end_anth, by = "uid")
  #####new filtering/joining ########

  data_old <- haven::read_dta("p1y2_endline_field_hh_PUBLIC.dta")
  data_2 <- data_old %>%
    mutate(
      happy_numeric = as.numeric(happy_likert),
      edu_numeric = as.numeric(resp_edu),
      ration_status = as.numeric(rationcard)
    ) %>%
    dplyr::select(uid, happy_numeric, edu_numeric, ration_status) %>%
    filter(!is.na(happy_numeric), !is.na(edu_numeric), !is.na(ration_status))

  p1y1_end_anth_old <- haven::read_dta("p1y1_endline_anthropometrics.dta")
  p1y1_end_anth <- p1y1_end_anth_old %>%
    mutate(whz_numeric = as.numeric(whz)) %>%
    dplyr::select(uid, whz_numeric) %>%
    filter(!is.na(whz_numeric)) %>%
    distinct(uid, .keep_all = TRUE)

  data <- data_2 %>%
    inner_join(p1y1_end_anth, by = "uid") %>%
    mutate(
```

```
      ration_status = factor(ration_status,
                             levels = c(1, 2, 3),
                             labels = c("APL", "BPL", "AAY"))
    ) %>%
    filter(!is.na(ration_status))



##### function ####### ####### ####### ####### ####### ####### ####### #######

dat = data
K = 5
target_edu = 8
target_happy = 4
target_ration = "APL"
my_new_knn <- function(
    K, target_edu, target_happy, target_ration, weights = c(1,1,1), dat
){
  tmp <- tibble(
    edu = c(target_edu, dat$edu_numeric),
    happy = c(target_happy, dat$happy_numeric),
    ration = factor(c(as.character(target_ration), as.character(dat$ration_status)))
  )


  gower_dist <- as.matrix(cluster::daisy(
    tmp, metric = "gower", weights = weights
  ))[1,-1]

  # now the usual knn process
  dat %>%
    ungroup %>%
    mutate(
      dist = gower_dist
    ) %>%
    arrange(dist) %>%
    slice(1:K) %>%
    dplyr::summarise(
      pred = mean(whz_numeric, na.rm = TRUE)) %>%
    dplyr::pull(pred)
}


# loop through a grid and make predictions
```

```r
edu_vector <- seq(
  min(as.numeric(data$edu_numeric), na.rm = T),
  max(as.numeric(data$edu_numeric), na.rm = T),
  by = 1
)



happy_vector <- seq(
  min(as.numeric(data$happy_numeric), na.rm = T),
  max(as.numeric(data$happy_numeric), na.rm = T),
  by = 1
)


ration_vector <- levels(data$ration_status)


grid <- expand.grid(
  edu = edu_vector,
  happy = happy_vector,
  ration = ration_vector,
  KEEP.OUT.ATTRS = FALSE,
  stringsAsFactors = FALSE
)


#################### testing ########################################
set.seed(12132005)
data_subset <- data %>% sample_n(400)
training_ndx <- sample(1:nrow(data_subset), size = round(.7*nrow(data_subset))) %>% sort()
testing_ndx <- c(1:nrow(data_subset))[-training_ndx]
training_df <- data_subset %>% ungroup %>% slice(training_ndx)
testing_df <- data_subset %>% ungroup %>% slice(testing_ndx)

# fit our KNN algorithm for multiple values of K
# predict the values of our testing data set for each value of K
# see which one does best
grid <- testing_df %>%
  ungroup %>%
  dplyr::select(edu = edu_numeric, happy = happy_numeric, ration = ration_status)

# K = K_values[k]
K_values <- c(3,5,10,20,40,60,80,100)
```

```
#dat = training_df
pred_matrix <- matrix(NA, nrow(grid), ncol = length(K_values))
for(k in 1:length(K_values)){
  data_subset <- data %>% sample_n(400)


  for(i in 1:nrow(grid)){
    pred_matrix[i,k] <- my_new_knn(
      K = K_values[k],
      as.numeric(grid[i,1]),
      as.numeric(grid[i,2]),
      as.numeric(grid[i,3]),
      weights = c(1,1,1),
      training_df
    )
  }
}

pred_tbl <- grid %>%
  mutate(
    truth = testing_df$whz_numeric,
    `3` = pred_matrix[,1],
    `5` = pred_matrix[,2],
    `10` = pred_matrix[,3],
    `20` = pred_matrix[,4],
    `40` = pred_matrix[,5],
    `60` = pred_matrix[,6],
    `80` = pred_matrix[,7],
    `100` = pred_matrix[,8]

  ) %>%
  pivot_longer(`3`:`100`, names_to = "K", values_to = "pred")

# compute our loss function
## attempt 2
pred <- pred_tbl %>%
  mutate(SE = (truth - pred)^2) %>%
  group_by(K)%>%
  summarise(mse = mean(SE))%>%
  arrange(mse)
```

```
data%>%
  ggplot(aes(x = edu_numeric, y = whz_numeric, colour = happy_numeric))+
  facet_wrap(~ration_status)+
```

```
  geom_point(, alpha = 0.6) +
  geom_smooth(method = "gam", formula = y ~ s(x, k = 5), se = TRUE, aes(group = 1))+
  labs(
    title = "Education Level versus WHZ by Happiness Score and Ration Status",
    x = "Education Level (Years)",
    y = "Weight to Heigh (Z-Score)"
  )+
  theme_bw()
```