Predicting Birth Weight

Nataniel Moreau

2023-07-27

Part 01

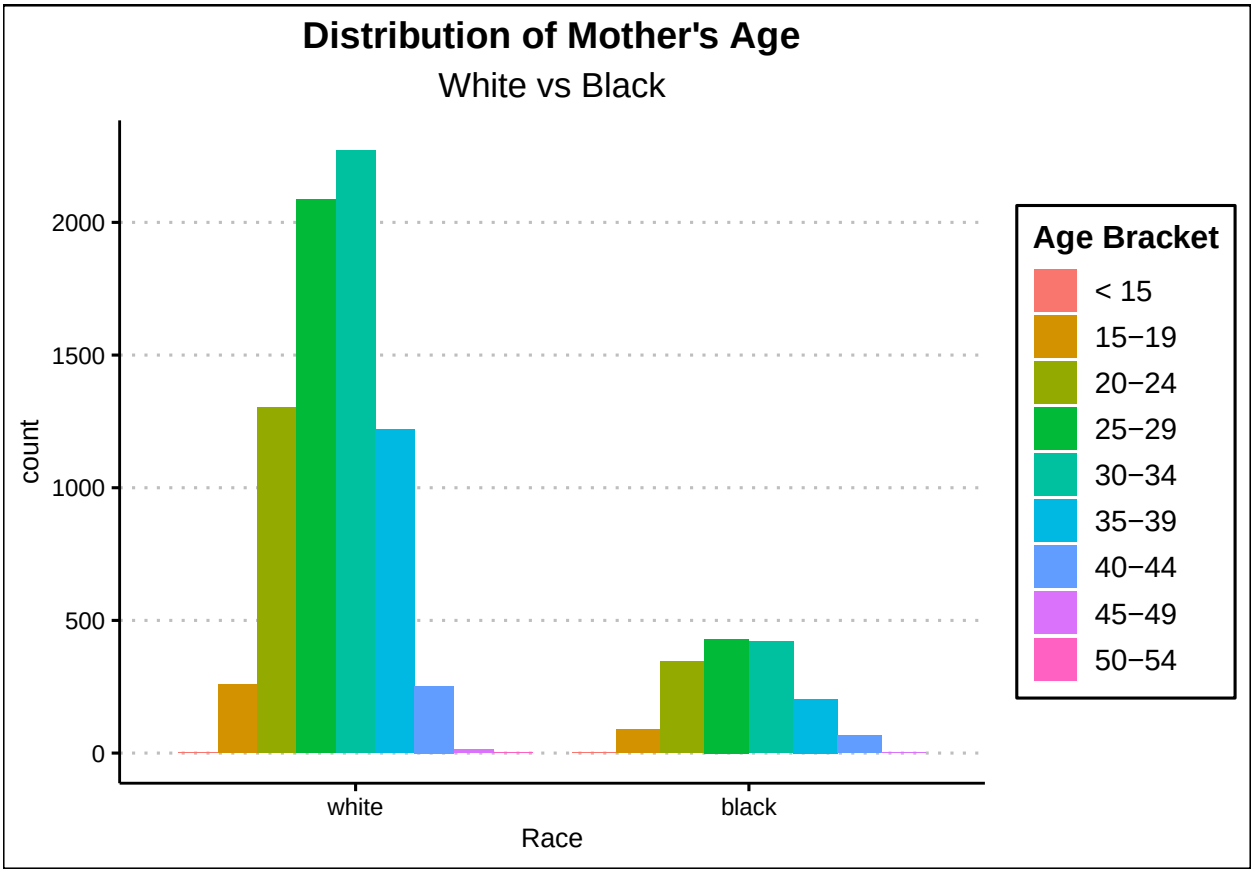
```
# Prep & clean & explore data ---------------------------------------------------------------------

library(pacman)
p_load(tidyverse, ggthemes, scales, tidymodels, janitor,
       magrittr, glmnet, modeldata,
       baguette, data.table, parallel, xgboost, skimr,
       scales, caret, leaps, MASS, usemodels)

#load data
birth_data = read_csv("data-final.csv")
```

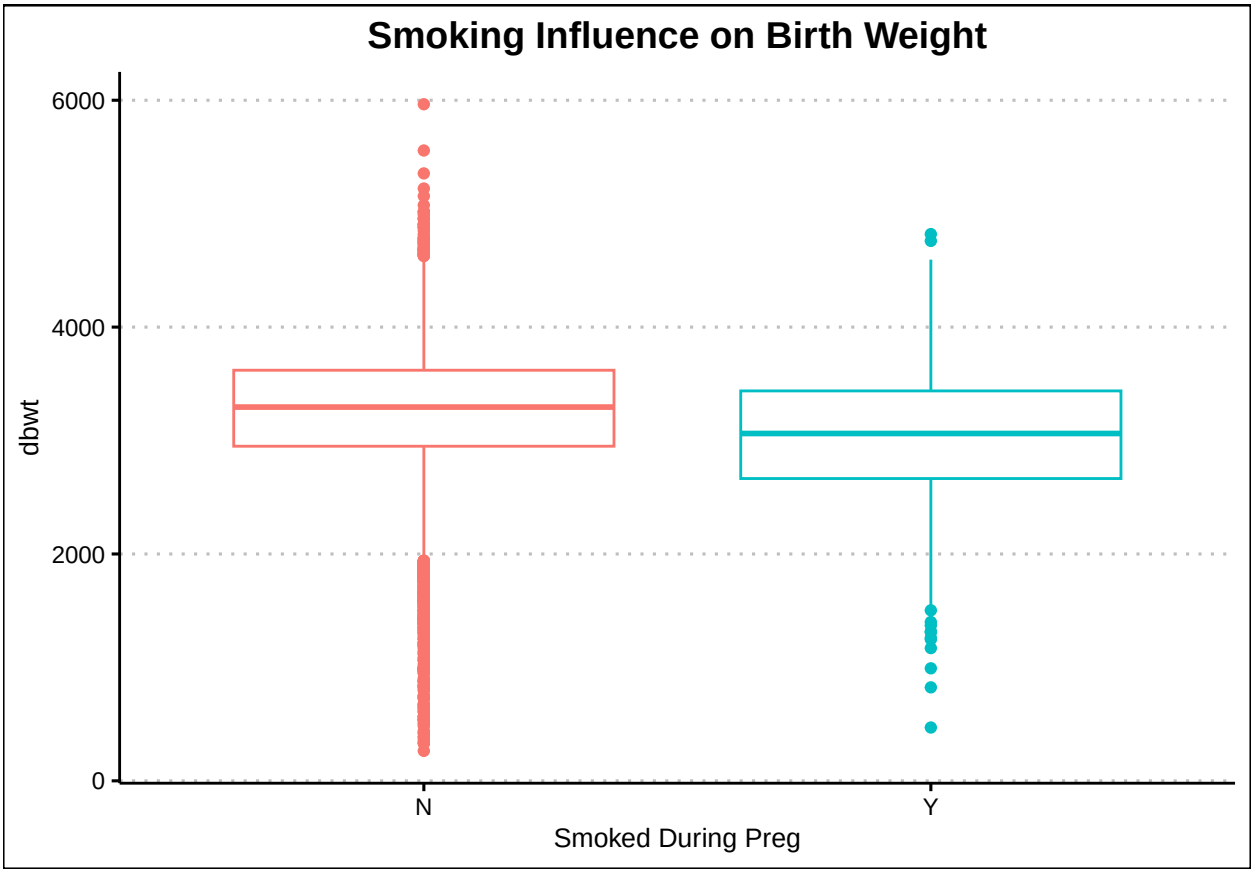
```
## Rows: 10000 Columns: 225
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (55): mar_p, wic, cig_rec, rf_pdiab, rf_gdiab, rf_phype, rf_ghype, rf_e...
## dbl (168): dob_yy, dob_mm, dob_tt, dob_wk, bfacil, f_facility, bfacil3, mage...
## lgl   (2): mage_repflg, imp_sex
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# mother's ages across races
birth_data %>% filter(mrace6 == 10| mrace6 == 20) %>%
  mutate(mrace6 = as_factor(mrace6),
                    mager9 = as_factor(mager9)) %>%
  ggplot(aes(mrace6, fill = mager9)) +
  geom_bar(stat = "count", position = "dodge") +
  labs(title = "Distribution of Mother's Age",
       subtitle = "White vs Black",
       x = "Race")+
  scale_x_discrete(labels = c("10" = "white", "20" = "black")) +
  scale_fill_discrete(name = "Age Bracket", labels = c("< 15", "15-19", "20-24", "25-29", "30-34",
                                                       "35-39", "40-44", "45-49", "50-54")) +
  theme_clean() +
  theme(plot.title = element_text(hjust = .5),
        plot.subtitle = element_text(hjust = .5))
```



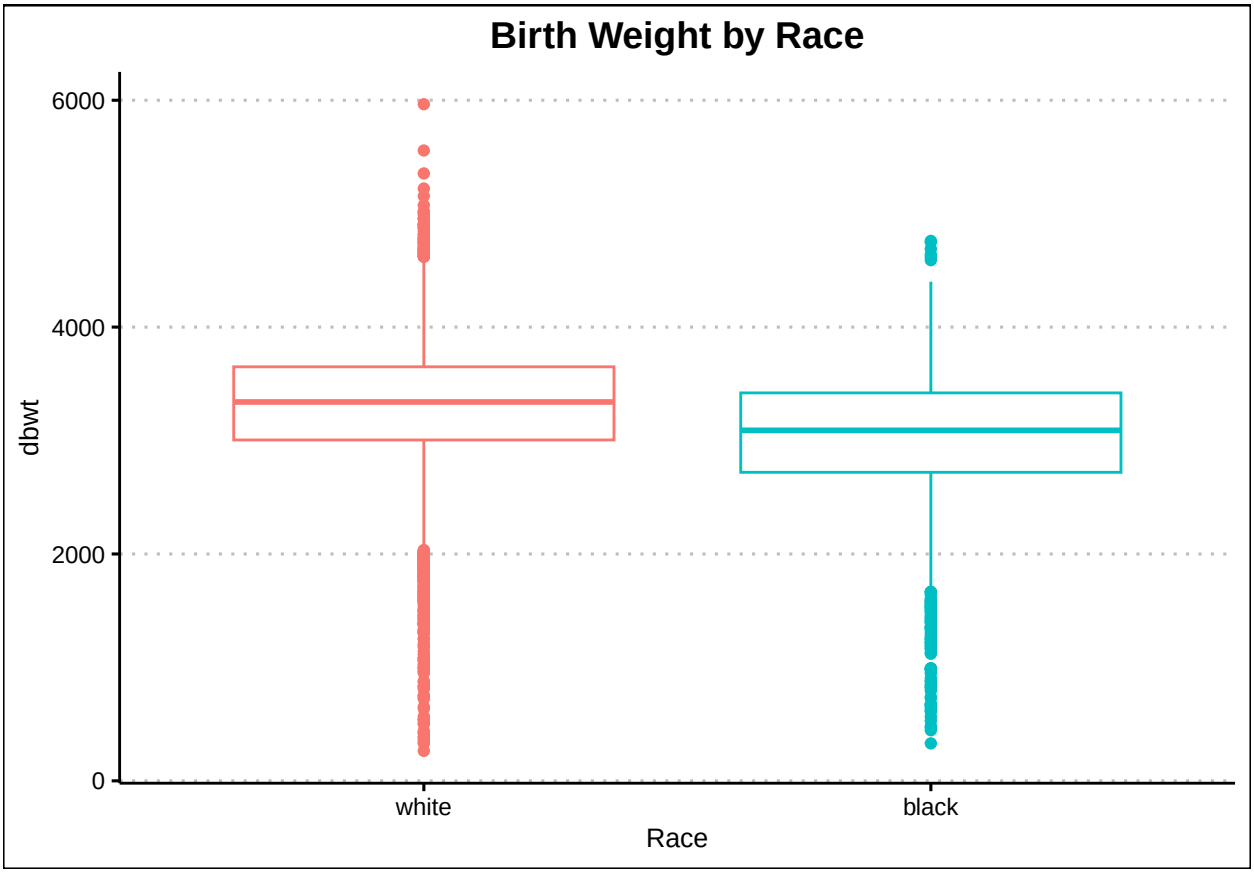
The ages that the women are baring children seem to be evenly distributed across the white and black populations. I would not have been suprised if black women were having chidlren earlier due to differences in income, sexual education, and access to affordable contraception and abortion.

```
# smoking influence on birth weight
birth_data %>% filter(dbwt < 7000,
                      cig_rec != "U") %>%
  ggplot(aes(cig_rec, dbwt,color = cig_rec)) +
  geom_boxplot() +
  labs(title = "Smoking Influence on Birth Weight",
       x = "Smoked During Preg") +
  theme_clean() +
  theme(legend.position = "none",
        plot.title = element_text(hjust = .5))
```



Somewhat surprisingly, smoking seems to have little effect on the average birth weight of the enfants. I expected to see sharper negative effects.

```
# birthweights difference black white
birth_data %>% filter(mrace6 == 10 | mrace6 == 20,
                      dbwt < 7000) %>%
  mutate(mrace6 = as.factor(mrace6))%>%
  ggplot(aes(mrace6, dbwt, color = mrace6)) +
  geom_boxplot()+
  labs(title = "Birth Weight by Race",
       x = "Race") +
  theme_clean() +
  scale_x_discrete(labels = c("10" = "white", "20" = "black")) +
  theme(legend.position = "none",
        plot.title = element_text(hjust = .5))
```



Similarly, birth weights are nearly identical across black and white populations.

Part 02: UNPENALIZED LINEAR REGRESSION

```
# Linear Model ------------------------------------------------------------
# cross validation
lin_cv = trainControl(method = "cv", number = 5)
# define model with chosen vars
lin_mod = train(dbwt ~ mager + as_factor(mrace6) + as_factor(frace6) + rf_gdiab + no_risks + no_infec,
                data = birth_data, method = "lm", trControl = lin_cv)
print(lin_mod)
```

```
## Linear Regression
##
## 10000 samples
##     6 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 8000, 7999, 8001, 8000, 8000
## Resampling results:
##
##   RMSE      Rsquared    MAE
##   617.4345  0.03579942  438.7766
```

```
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

This model does not seem particularly great. An rmse of 600 is not very accurate and is three times as large as my best model. I don't think that I did a very good job of choosing the variables, as my R2 was only ~4%. Again making sense logically to a human brain doesn't make a a variable a good predictor.

Part 03: LASSO

```
# Lasso ------------------------------------------------------------------
# cv
birth_cv = birth_data %>% vfold_cv(v = 5)

# define recipe
birth_recipe = recipe(dbwt ~ ., data = birth_data) %>%
  # remove vars with only 1 value b4 normalization
  step_zv(all_predictors()) %>%
  # deal with NAs
  step_impute_mode(all_nominal_predictors()) %>%
  step_impute_median(all_numeric_predictors())%>%
  step_normalize(all_numeric_predictors()) %>%
  # dummy factor levels
  step_dummy(all_nominal_predictors())

# define model
lasso_mod = linear_reg(penalty = tune(), mixture = 1) %>%
  set_engine("glmnet")

# set workflow
lasso_wkfl = workflow() %>%
  add_model(lasso_mod) %>%
  add_recipe(birth_recipe)

set.seed(246810)

# fit model
lasso_fit = lasso_wkfl %>% tune_grid(birth_cv, grid = expand_grid(penalty = seq(.1,5,by = .1)),
                                     metrics = metric_set(rmse))
# select final model
show_best(lasso_fit, metric = "rmse")
```

```
## # A tibble: 5 x 7
##   penalty .metric .estimator  mean     n std_err .config
##     <dbl> <chr>   <chr>      <dbl> <int>   <dbl> <chr>
## 1     5   rmse    standard    176.     5    12.8 Preprocessor1_Model50
## 2     4.9 rmse    standard    176.     5    12.7 Preprocessor1_Model49
## 3     4.8 rmse    standard    176.     5    12.7 Preprocessor1_Model48
## 4     4.7 rmse    standard    176.     5    12.7 Preprocessor1_Model47
## 5     4.6 rmse    standard    176.     5    12.7 Preprocessor1_Model46
```

```
lasso_final =
  lasso_wkfl %>%
  finalize_workflow(select_best(lasso_fit, metric = "rmse"))
```

The penalized model blew the standard OLS model out of the water with a rmse of ~170. It makes sense as there are a lot of vars that seem like nonsense in this data set so having steep penalty (2) should increase performance and give a slim model. The model did not choose the same variables as I did, which is to be expected given the difference in performance.

Part 04: BOOSTED TREES

```
boost_mod = boost_tree(mode = "regression",
                           mtry = tune(),
                           min_n = 2,
                           trees = 100,
                           tree_depth = tune(),
                           learn_rate = tune()) %>%
  set_engine("xgboost")

# deine workflow
boost_wrkfl = workflow() %>%
  add_model(boost_mod) %>%
  add_recipe(birth_recipe)

set.seed(246810)

#fit model
boost_fit = boost_wrkfl %>%
  tune_grid(birth_cv, grid = expand_grid(mtry = seq(6,10, by = 2),
                                         tree_depth = seq(6,10, by = 2),
                                         learn_rate = seq(.05,.1, by = .01)),
            metrics = metric_set(rmse))

#select final model
show_best(boost_fit, metric = "rmse")
```

```
## # A tibble: 5 x 9
##    mtry tree_depth learn_rate .metric .estimator  mean     n std_err .config
##   <dbl>      <dbl>      <dbl> <chr>   <chr>      <dbl> <int>   <dbl> <chr>
## 1    10          8       0.09 rmse    standard    239.     5    24.1 Preprocess~
## 2    10         10       0.1  rmse    standard    242.     5    24.9 Preprocess~
## 3    10         10       0.09 rmse    standard    244.     5    25.0 Preprocess~
## 4     8         10       0.1  rmse    standard    247.     5    28.4 Preprocess~
## 5    10         10       0.08 rmse    standard    252.     5    25.5 Preprocess~
```

```
final_boost =
  boost_wrkfl %>%
  finalize_workflow(select_best(boost_fit, metric = "rmse"))
```

My boosted trees model split the difference performing better than the simple linear model but worse than my LAssO. Increasing non-linearity does not appear to help in this setting. The model was relatively quick with fast learning rates and deep trees.

Part 05: SUMMARY

My LASSO model definitely performed the best, it had a lower estimated test error and ran significantly faster than the boosted trees model. The rmse is significantly better by ~30%. The relatively steep penalty and success of the linear model over the boosted trees model would suggest that the relationship between birth weight and its covariates is a relatively straight forward one, at least in terms of the important interactions. The process of finding those important interactions remains best left to tuning and cross validation.