Final Report: Predicting Cancer Mortality On County-Level Data

Jasen Zhang, Alexander Zhu, Christian Pascual

1 Introduction

Cancer ranks among the leading causes of death worldwide. According to a report published by the American Cancer Society, 1.8 million new cancer cases are estimated to be diagnosed in 2020 in the United States, and more than 600,000 are expected to die directly as a result of cancer [1]. Cancer also carries a significant economic burden, costing the United States an estimated $80.2 billion according to a 2015 report [2]. From both a humane and economic perspective, investigation into effective cancer interventions represents a important front of research.

A wide berth of research has demonstrated that environmental and socioeconomic factors contribute to increased risk of cancer and mortality. For example, Adler et. al showed that higher socioeconomic status was associated with decreased cancer mortality [3]. Rawl et. al demonstrates a similar result in a statewide survey in Indiana that income and education were inversely related to cancer mortality. Rawl also discusses how race affected cancer mortality in their sample, finding that African-American participants worried less about cancer and were less likely to seek treatment [4]. Rohfling et. al found that uninsured patients or those under Medicaid were more likely to have more advanced tumors and poorer survival compared to peers with private insurance [5]. Higher socioeconomic status gives people better access to healthcare resources that are potentially life-saving or will help increase survival, and the opposite effect has been seen the lower scale.

Similar research has documented race-based health disparities in cancer mortality. In an observational study in Philadelphia, Zeigler-Johnson found that black men were at the highest risk of prostate cancer relative to similar white counterparts [6]. Looking at data spanning from 1950 to 2014, a study by Singh showed that individuals from lower educational backgrounds experienced higher mortality of various types of cancer. Furthermore, African-Americans saw higher cancer mortality compared to their Asian and White counterparts in this group [7]. These are just a few examples that have documented how race influences cancer mortality, illustrating that it is a critical factor to consider race-based health disparities in any cancer-related intervention.

One difficulty in researching cancer is that it is an incredibly diverse collection of diseases, as opposed to a single monolithic illness. Further complicating this is that different cancers can occur at different rates across different regions of the United States. Mokdad et. al found that there were distinct clusters of counties in different regions with especially high cancer mortality. For example, breast cancers are highly prevalent in the southern belt, whereas liver cancer is the prevailing diagnosis along the Texas-Mexico border [8]. Just as the aforementioned studies examined how socioeconomic factors and race can influence cancer mortality on an individual level, it could be useful to understand how these factors relate to cancer mortality on a wider geographic level. Understanding how the relationship between these factors and cancer mortality on a geographic level offers an opportunity for researchers to create predicive models that can help us see what

areas might have higher cancer burdern and thus be a way for us to know how we might to allocate limited resources to areas that are harder hit.

1.1 Data

For our analysis, we used a dataset aggregated from multiple sources, noted later. The data spanned from 2010 to 2016 and included information on 3047 counties in the United States, the county being the observational unit in the data. There were a total of 3,143 counties and county-equivalents in the United States, so 3.1% of them were represented. Our group did not aggregate the data ourselves, it is publically available and can be found at this link. Table 1 contains a full overview of the data:

	Mean	Std	R_Squared
avgAnnCount	589.90	1173.10	-0.167
avgDeathsPerYear	179.65	408.59	-0.102
TARGET_deathRate	178.52	27.50	1.000
incidenceRate	447.35	51.47	0.442
medIncome	47109.68	12108.60	-0.434
popEst2015	98482.89	260559.49	-0.140
povertyPercent	16.87	6.44	0.436
studyPerCap	157.89	537.09	-0.022
MedianAge	40.83	5.20	-0.008
MedianAgeMale	39.57	5.24	-0.027
MedianAgeFemale	42.15	5.30	0.011
AvgHouseholdSize	2.53	0.25	-0.037
PercentMarried	51.82	6.88	-0.270
PctNoHS18_24	18.20	8.04	0.093
PctHS18_24	35.00	9.11	0.263
PctSomeCol18_24	40.62	11.00	-0.166
PctBachDeg18_24	6.17	4.55	-0.289
PctHS25_Over	34.78	7.03	0.410
PctBachDeg25_Over	13.31	5.42	-0.488
PctEmployed16_Over	54.17	8.34	-0.417
PctUnemployed16_Over	7.82	3.45	0.382
PctPrivateCoverage	64.39	10.63	-0.394
PctPrivateCoverageAlone	48.49	10.07	-0.374
PctEmpPrivCoverage	41.21	9.45	-0.270
PctPublicCoverage	36.24	7.87	0.408
PctPublicCoverageAlone	19.23	6.14	0.453
PctWhite	83.73	16.35	-0.181
PctBlack	9.01	14.51	0.263
PctAsian	1.25	2.62	-0.187
PctOtherRace	1.99	3.54	-0.191
PctMarriedHouseholds	51.28	6.56	-0.300
BirthRate	5.65	1.99	-0.091

Table 1: Data summary

Our data contained information on various demographic, socioeconomic, household, and cancer-

related factors for each county, represented typically as percentages. These data were taken from the 2013 Census. In terms of race-related variables, twe had percentages of each county that identified as a particular race (Asian, Black, White and Other). The data also contained information on educational acheivement as well, measured as the proportion of the county population who have achieved a certain degree of education (high school and college). The education variables were also divided up by age, with separate columns for people ages 18 to 24 and another set 25 years and over. The proportion of people who have public and private health insurance was another notable variable in the data. A few variables in the dataset span many magnitudes, like median income, so we considered performing a log transform as we chose features to inclide in the final model.

The cancer-related variables data were aggregated from the American Community Survey, cancer.gov and clinicaltrials.gov, spanning from 2010 to 2016. In terms of important cancer-related factors, we had the incidence rate of *all* cancer diagnoses in the county, measured in terms of *mean per capita (100,000 people)* and the average number of cancer cases reported annually from 2010 to 2016. Our target response was cancer mortality, also measured as mean per capita. In this dataset, cancer mortality ranged from 59.7 to 362.8, with a median value of 178.1.

1.2 Initial Analyses & Objectives

Given our literature and what is present in our dataset, we aimed to examine how different socioeconomic and demographic factors are associated with cancer mortality on a county level.

1.2.1 Correlation Between Predictors

Many of the predictors were highly correlated, so we created a heat map to keep track of these intercorrelations. Figure 1 below shows this heat map.

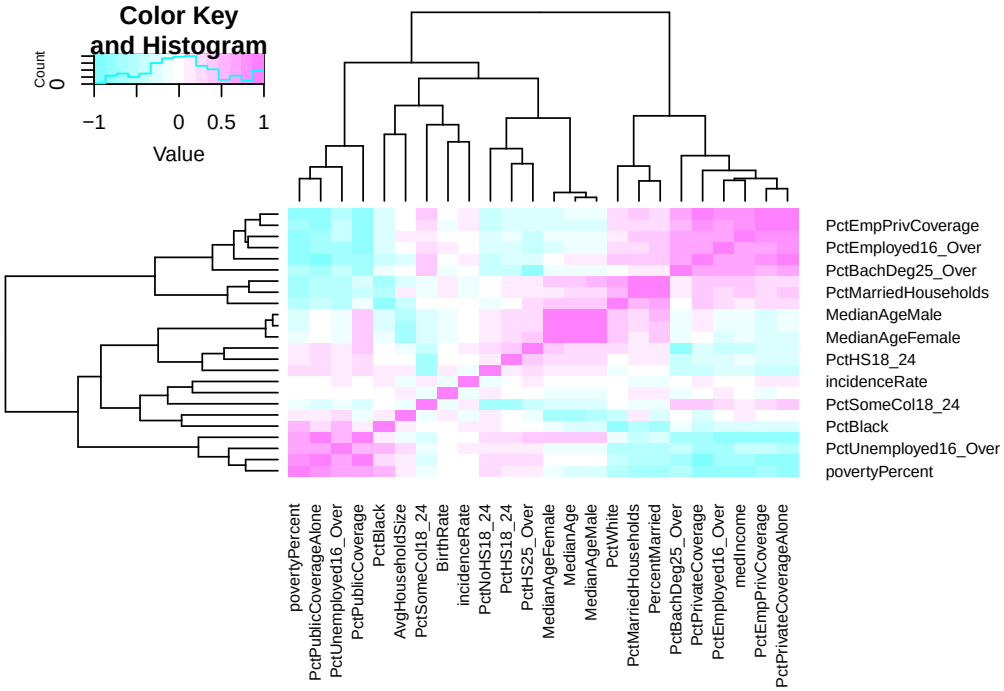

Figure 1: Correlation heatmap between variables in our cancer mortality dataset.

The percentage of poverty in a county correlated highly with other predictors that deal with being

under public insurance or being unemployed. The median ages of men and women were highly correlated as well. The education variables are also highly correlated. With these correlations in mind, we know that we need to account for this and include only one of them.

1.2.2 Interaction Between Race & Education

One interesting trend that we found in the data was that there seemed to be an interaction between race and education in relation to cancer mortality. According to an article by the US Department of Education, the median percentage of adults 25 and over completing a bachelor's degree was 34%. We divided the counties by if they fell below or were equal to higher to the median value and investigated how cancer mortality changed with percentage of race (Figure 2).

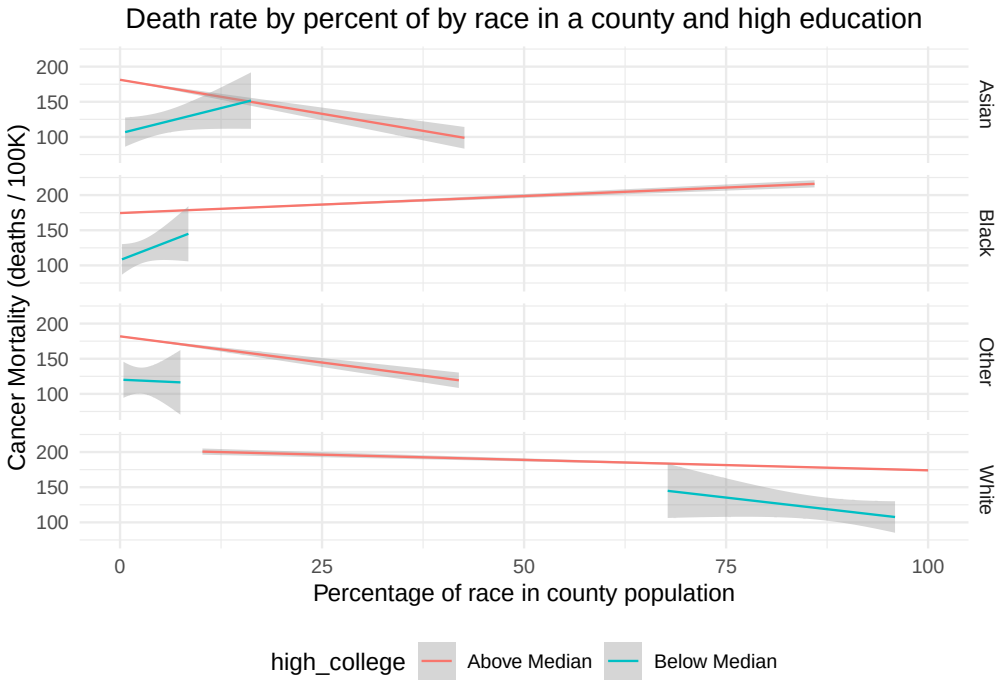

Figure 2: Effects plot of percentage of race in county, stratified by education achievement

The change is most drastic in Asians, but the trend holds over all races present in the data. For African-Americans, we also see an attenuation of the average cancer mortality desite it not converting to a negative correlation. Our literature review showed that both education and race are both significantly associated with cancer mortality, so it raise an interesting question on if this can be utilized on a county level.

Based on the results of our exploratory analyses, we propose two analytic questions:

1. Can we identify county-level factors that contribute to a significant difference in cancer mortality? If so, can we identify any significant interaction between these factors that also contribute to increased cancer mortality?
2. Can we create an effective predictive model from the data? If we can find any significant interactions from our explanatory model, will they be helpful in increasing predictive ability?

The focus of our project will be to create a useful predictive model for cancer mortality.

2 Methodology

2.1 Explanatory Model

2.1.1 Model Selection

With 30 covariates, we tried a few different candidate models before deciding on a final model. As seen in our correlation heatmap, we decided on a subset of variables to use instead of creating combined versions. Through a series of ANOVA, we found that an interaction model contained significant interaction between education and race produced the best fit in terms of adjusted R^2, so this became a focal point of our analysis. The education variables focused on ages 25 and above created better model fit than the

Our final explanatory model is as follows:

$$Y_i = X_{education}\beta + X_{race}\beta + X_{interaction}\beta + X_{confounders}\beta + \epsilon_i$$

In terms of education variables, we chose 2 from the entire group: 1) "percent of the county ages 25 and over finishing high school", and 2) the percent over 25 finishing college (bachelor's degree). For our race variables, we included all that were present in the data (% of the county being Asian, Black, White or Other Race). With our literature review and prior knowledge, we knew that age, sex, cancer incidence, income, and population need to be accounted for in the model since they are known confounders between cancer mortality and our covariates of interest.

2.1.2 Analysis Plan

As mentioned before, the data concerning education and race come from the 2013 Census. We know the the census data comes from a carefully picked representative population in each county, but this information is not available in the data itself. Despite the data containig most of the counties in the United States, the randomness introduced by the survey requires us to perform some inference. We expect there to be violations to typical regression assumptions since adjacent counties may be correlated. To account for this, we plan to use 1000 bootstrap samples to calculate a robust confidence interval for each of the coefficients. Coefficients that we find to be significant in the explanatory model will be included in a prediction model candidate.

2.2 Prediction Models

We built three models to predict county cancer death rates using both the raw explanatory variables and interaction variables we found to be significant. We built a LASSO regularization model, principal component analysis (PCA) model, and stepwise selection model. We randomly split the data into 80% train and 20% test and used this same split for all three models. The LASSO and PCA models had hyperparmeters to optimize, so these were optimized using 10 fold cross-validation on the training set. Choosing 10 folds and an 80-20 split are conventional in machine learning models, so we follow this best practice.

2.2.1 LaSSO

In the LASSO model, the aim is to minimize not just the residual sum of squares (RSS), but the sum of both RSS and a scalar multiple (λ) of the absolute sum of the beta coefficients. This extra penalty term allows for the moderation of beta coefficients and serves as another layer of feature

selection. We performed 10 fold cross-validation to select for the optimal value of λ, and then fit the LASSO model on the entire training set. The resulting model beta was then used to predict cancer mortality in the test set.

2.2.2 PCA

In the PCA model, we used 10 fold cross validation to select for an optimal number of principal components (PC). Since the error metric we used was root mean square error (RMSE), we chose to follow a elbow heuristic to choose the optimal number of PCs. The elbow rule selects a number of PCs that explains an appreciable amount of the variance while minimizing the diminishing returns from choosing more PCs when adding additional PCs Once we selected the optimal number of PCs, 10, we computed the first 10 PC loadings and used those loadings to predict the cancer death rates in the train and test sets.

2.2.3 Stepwise Selection

We used the stepwise selection algorithm with AIC as a guiding metric on the entire training set to select for a well-fitting set of variables. These variables were then fitted on the training set and used to predict the cancer death rates in the test set.

2.2.4 Model Metrics

To assess each of these models, we computed the RMSE and R^2 values of the true cancer death rates versus predicted cancer death rates of the training set. The model with the highest RMSE and/or R^2 values would then be the ideal model candidate to perform the test set predictions.

3 Results & Discussion

3.0.1 Initial Explanatory Model

Table 2 shows the initial estimates for our final explanatory model. Looking at the main effects, we see that with the exception of Asians, higher proportions of the other races was associated with higher average cancer mortality in a county. Most of the interaction terms are small and insignificant, but most are associated with average decreases in mortality.

Table 2: Estimated coefficients and 95% CI

term	estimate	95% CI
pct_hs25_over	2.870	(2.26, 3.48)
pct_bach_deg25_over	2.066	(0.953, 3.179)
pct_white	1.164	(0.852, 1.476)
pct_black	1.061	(0.71, 1.412)
pct_asian	-0.589	(-1.927, 0.749)
pct_other_race	-1.507	(-2.196, -0.819)
pct_hs25_over:pct_white	-0.026	(-0.032, -0.019)
pct_hs25_over:pct_black	-0.032	(-0.039, -0.025)
pct_hs25_over:pct_asian	0.017	(-0.011, 0.045)
pct_hs25_over:pct_other_race	0.014	(-0.002, 0.03)
pct_bach_deg25_over:pct_white	-0.041	(-0.052, -0.029)

term	estimate	95% CI
pct_bach_deg25_over:pct_black	-0.002	(-0.014, 0.01)
pct_bach_deg25_over:pct_asian	-0.006	(-0.039, 0.027)
pct_bach_deg25_over:pct_other_race	0.029	(0.007, 0.052)

Looking at our regression diagnostics, we confirmed that the errors did not seem to come from a normal distribution based on deviations in the QQ plot.

Figure 3: Effects plot of percentage of race in county, stratified by education achievement

3.0.2 Interaction Between Education & Race

Figure 3 shows the 95% bootstrap percentile intervals of 1000 bootstrap interaction models. The bootstrap 95% confidence intervals indicate that only the high school interactions with African- and white Americans were significant. The coefficients associated with these interactions are all negative, which matched with our original estimated model.

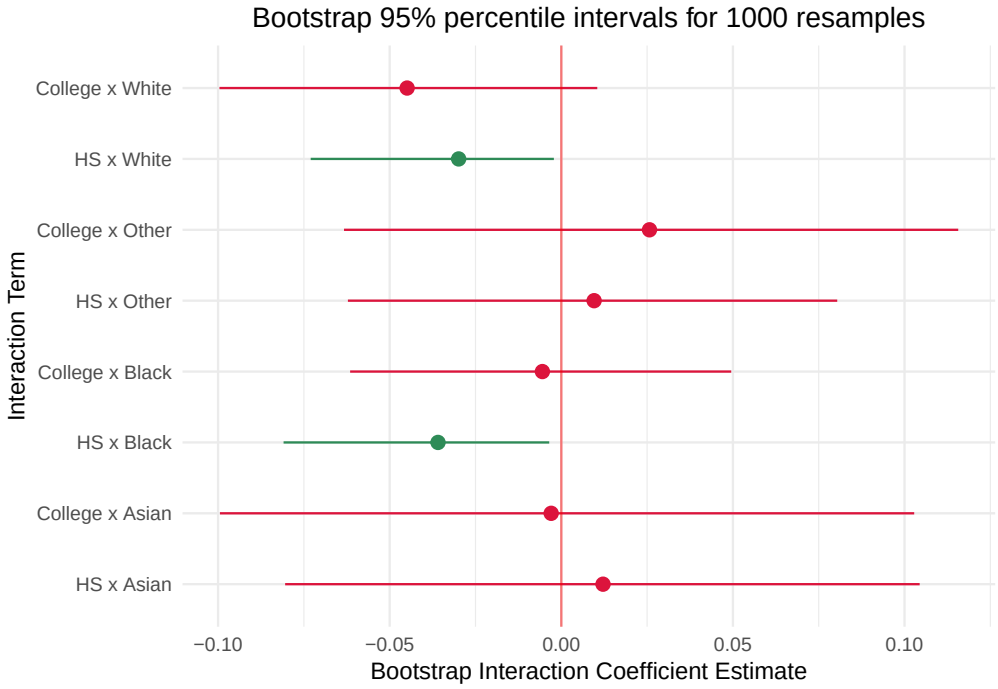

Figure 4: 95

With these results and literature review in mind, we added HS x White, HS x Black and BS Black to the prediction models

3.1 Prediction Model

The LASSO model selected an optimal hyperparameter of $\lambda = 0.00119$ as shown in Figure 5a. The estimated coefficients fitted from the training set are shown in Table Y, with only the intercept and incidence rate having coefficients of 0. The training set RMSE and R^2 was 13.8 and 0.866 respectively, and the test set metrics showed slight improvements with an RMSE and R^2 of 12.8 and 0.872 respectively.

Using the elbow rule as shown in Figure 6a, we selected 10 PCs to be the optimal amount and obtained the loadings of the first 10 PCs from the training set. Roughly half of the variables did not show up significantly in any of the 10 PCs. The training set RMSE and R^2 was 20.0 and 0.689 respectively, and the test set metrics showed moderate improvements with an RMSE and R^2 of 17.9 and 0.728 respectively.

Our stepwise selection model included 23 variables, and the order in which they were selected is displayed in Figure B. The estimated coefficients fitted from the training set are shown in Table Y along with the lasso regularization coefficients. The training set RMSE and R^2 was 13.2 and 0.877 respectively, and the test set metrics showed slight improvements with an RMSE and R^2 of 12.4 and 0.879 respectively. These values are in Supplemental Table Z. This model also excluded 2 of the interaction terms that were significant in the explanatory model.

	Lasso	PCA	Stepwise
R_squared_train	0.866	0.689	0.877
R_squared_test	0.872	0.728	0.879
RMSE_train	13.796	20.020	13.278
RMSE_test	12.842	17.955	12.495

Table 2: Training and test metrics for the 3 predictive models.

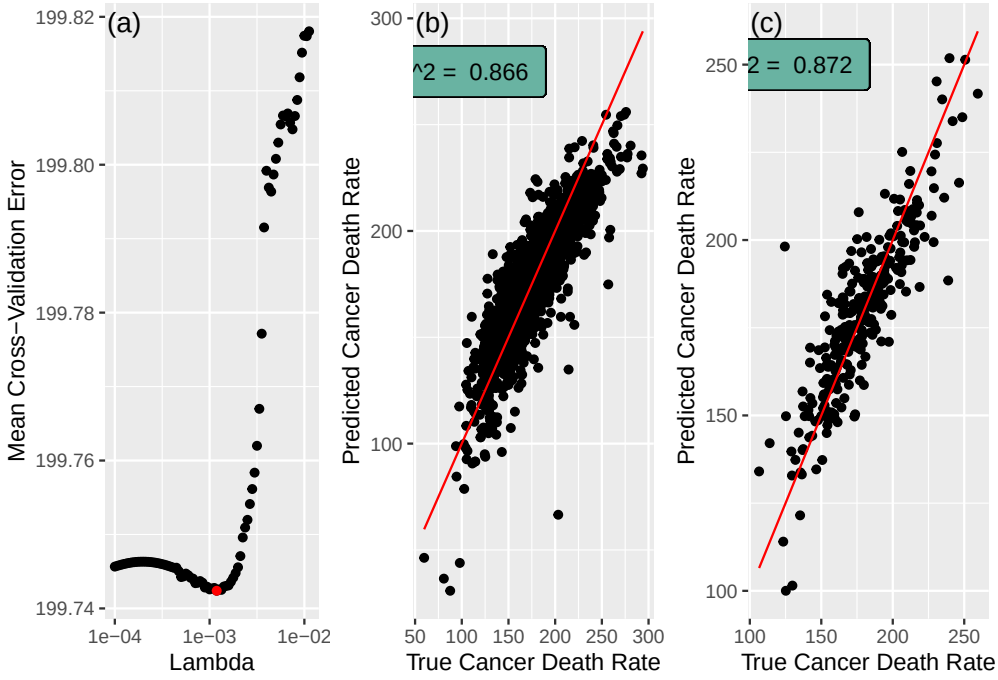

Figure 5: Hyperparameter Choice and train/test plots for LASSO model. a) hyperparameter optimization, b) fit on the training dataset, c) fit on the test dataset

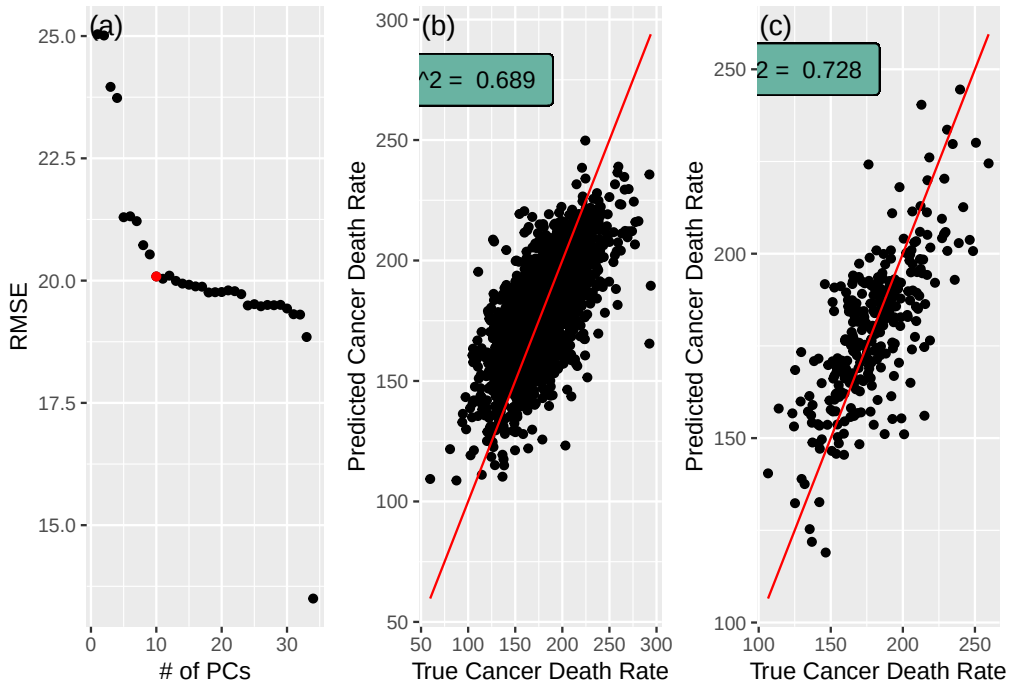

Figure 6: Hyperparameter Choice and train/test plots for PCA model. a) hyperparameter optimization, b) fit on the training dataset, c) fit on the test dataset

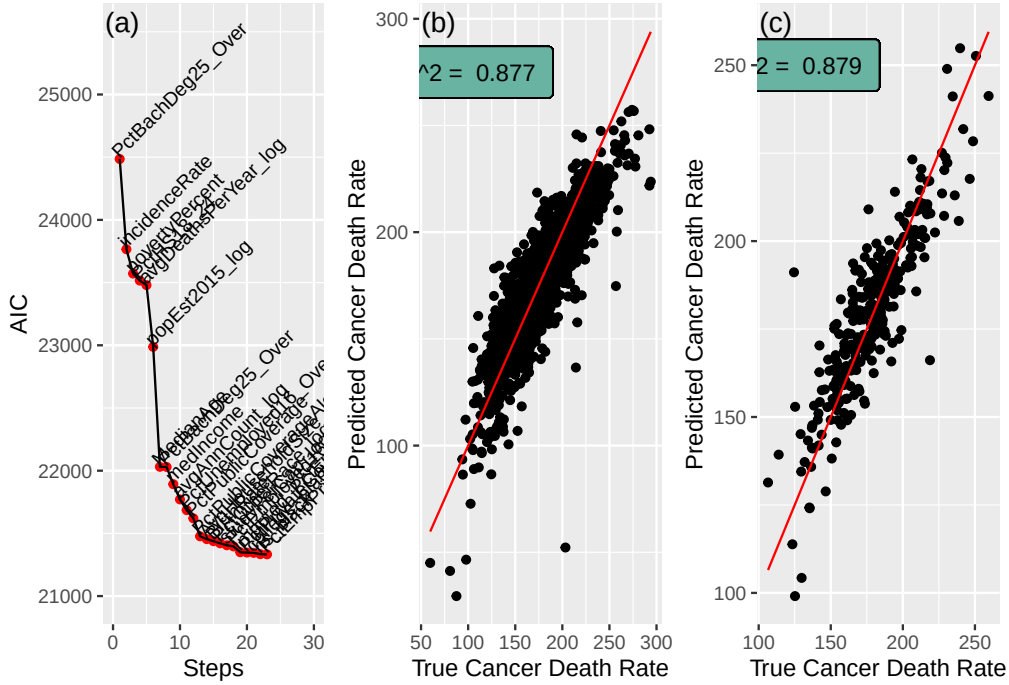

Figure 7: Hyperparameter Choice and train/test plots for stepwise model. a) AIC as features added to model, b) fit on the training dataset, c) fit on the test dataset

4 Conclusion

Our analysis found at least one significant interaction between race and education that served to help reduce cancer mortality per capita on a county level. However, this effect was small, and these interaction variables did not appear helpful in our stepwise model. Despite finding some significant terms, our predictive model showed that adding them was not particularly helpful. We considered using this explanatory model as a means of feature selection before we formally learned it in lecture, but we still thought it might be of value as a preliminary step for the prediction model.

The stepwise selection model showed the best RMSE and R^2 in the training set. We also see that the stepwise selection model had the best metrics for the test set. The stepwise selection model also included less features (24) than the LASSO model (33), so it's more parsimonious. Most variables in the stepwise selection model had statistically significant betas which are indicators of cancer mortality. The LASSO and stepwise models do not differ appreciably in terms of prediction error, so we ultimately opt for the stepwise model as the best candidate prediction model to use.

4.1 Contributions

- **Alexander**: created the table characterizing the data, boxplots for presentation, editing/revising
- **Jasen**: handled predicitve modeling, writing for predictive model parts of the report, editing/revising
- **Christian**: handled explanatory modeling, writing for explanatory model parts of the report, performed the literature review, editing/revising

Our project repo can be found at on Github (click me), commits can be reviewed through time.

5 References

1. https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2020/cancer-facts-and-figures-2020.pdf
2. https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2020/cancer-facts-and-figures-2020.pdf
3. ADLER, N.E. and OSTROVE, J.M. (1999), Socioeconomic Status and Health: What We Know and What We Don't. Annals of the New York Academy of Sciences, 896: 3-15. https://doi.org/10.1111/j.1749-6632.1999.tb08101.x
4. Rawl SM, Dickinson S, Lee JL, Roberts JL, Teal E, Baker LB, Kianersi S, Haggstrom DA. Racial and Socioeconomic Disparities in Cancer-Related Knowledge, Beliefs, and Behaviors in Indiana. Cancer Epidemiol Biomarkers Prev. 2019 Mar;28(3):462-470. doi: 10.1158/1055-9965.EPI-18-0795. Epub 2018 Nov 28. PMID: 30487135.
5. Rohlfing ML, Mays AC, Isom S, Waltonen JD. Insurance status as a predictor of mortality in patients undergoing head and neck cancer surgery. Laryngoscope. 2017 Dec;127(12):2784-2789. doi: 10.1002/lary.26713. Epub 2017 Jun 22. PMID: 28639701; PMCID: PMC5688011.
6. Zeigler-Johnson C, Keith S, McIntire R, Robinson T, Leader A, Glanz K. Racial and Ethnic Trends in Prostate Cancer Incidence and Mortality in Philadelphia, PA: an Observational Study. J Racial Ethn Health Disparities. 2019 Apr;6(2):371-379. doi: 10.1007/s40615-018-00534-z. Epub 2018 Dec 5. PMID: 30520002.
7. Singh, Gopal & Jemal, Ahmedin. (2017). Socioeconomic and Racial/Ethnic Disparities in Cancer Mortality, Incidence, and Survival in the United States, 1950–2014: Over Six Decades of Changing Patterns and Widening Inequalities. Journal of Environmental and Public Health. 2017. 1-19. 10.1155/2017/2819372.
8. Mokdad AH, Dwyer-Lindgren L, Fitzmaurice C, et al. Trends and Patterns of Disparities in Cancer Mortality Among US Counties, 1980-2014. JAMA. 2017;317(4):388–406. doi: 10.1001/jama.2016.20324
9. https://nces.ed.gov/pubs93/93442.pdf

6 Appendix

```
library(tidyverse)
library(knitr)
library(gplots)
library(gridExtra)
library(glmnet)
library(grid)
library(caret)
library(olsrr)
set.seed(1)
knitr::opts_chunk$set(
  fig.align = "center",
  out.width = "80%"
)
# Load the data and do some processing
cancer = read_csv("cancer_registry.csv") %>%
  # Split the geography variable
  separate(Geography, into = c("county", "state"), sep = ", ") %>%
  # Split up binnedInc into a lower and upper decile
  mutate(
    binnedInc = str_remove_all(binnedInc, "[(\\]]"),
    # also try to group states by region
    region = case_when(
      state %in% c("California", "Oregon", "Washington", "Nevada", "Idaho",
                   "Montana", "Wyoming", "Colorado", "Utah", "Alaska", "Hawaii") ~ "West",
      state %in% c("Arizona", "New Mexico", "Texas", "Oklahoma") ~ "Southwest",
      state %in% c("North Dakota", "South Dakota", "Nebraska", "Kansas",
                   "Minnesota", "Iowa", "Missouri", "Wisconsin", "Illinois",
                   "Indiana", "Ohio", "Michigan") ~ "Midwest",
      state %in% c("Arkansas", "Louisiana", "Mississippi", "Alabama", "Georgia",
                   "Florida", "South Carolina", "North Carolina", "Tennessee",
                   "Kentucky", "Virginia", "West Virginia", "District of Columbia",
                   "Delaware") ~ "Southeast",
      state %in% c("Maryland", "Pennsylvania", "New Jersey", "New York", "Rhode Island",
                   "Connecticut", "Massachusetts", "New Hampshire", "Vermont", "Maine") ~ "Nort
      TRUE ~ "Southwest" # Weird formatting means a single NM is NA in state
    )
  ) %>%
  separate(binnedInc, into = c("inc_dec_low", "inc_dec_high"), sep = ",") %>%
  janitor::clean_names() %>%  # Convert all column names to lowercase'
  mutate(
    high_college = pct_bach_deg25_over > median(pct_bach_deg25_over), # median(pct_bach_deg18_
    high_hs = pct_hs18_24 > median(pct_hs18_24)
  )
df = read_csv("cancer_registry.csv") %>%
  mutate(PctSomeCol18_24 = 100 - PctNoHS18_24 - PctHS18_24 - PctBachDeg18_24) %>% filter(incide
```

```
  filter(avgAnnCount < 20000) %>%
  filter(MedianAge < 200) %>%
  filter(AvgHouseholdSize > 1)
vars <- colnames(df)
misc_vars <- c('binnedInc', 'Geography', 'TARGET_deathRate')
vars_1 <- setdiff(vars, misc_vars)
response_vars <- c('TARGET_deathRate')
predict_vars <- paste(vars_1, collapse = ' + ')
df <- df %>% select(- c('binnedInc', 'Geography'))

df <- data.frame(sapply(df, as.numeric))
df2<-df[, c(colnames(df)[7],colnames(df)[9:32])]

column_names <- c()
res <- c()
outlier <- c()
for (i in colnames(df)){
  if (class(df[[i]]) == "numeric") {
     column_names <- c(column_names, i)
     mean <- round(mean(df[[i]], na.rm = TRUE),2)
     std <- round(sd(df[[i]],na.rm = TRUE),2)
     cor <- round(cor(df[[i]], df[['TARGET_deathRate']], use="complete.obs"),3)
     res <- c(res, mean)
     res <- c(res, std)
     res <- c(res, cor)
  }
}

output_table <- matrix(res, ncol=3,byrow=TRUE)
colnames(output_table) <- c("Mean", "Std", "R_Squared")
rownames(output_table) <- column_names
output <- as.table(output_table)
output %>%  kable()
df <- read.csv('cancer_registry.csv') %>%
  mutate(PctSomeCol18_24 = 100 - PctNoHS18_24 - PctHS18_24 - PctBachDeg18_24) %>%
  filter(incidenceRate < 1000) %>%
  filter(avgAnnCount < 20000) %>%
  filter(MedianAge < 200) %>%
  filter(AvgHouseholdSize > 1)
df <- na.omit(df)
vars <- colnames(df)
misc_vars <- c('binnedInc', 'Geography', 'TARGET_deathRate')
vars_1 <- setdiff(vars, misc_vars)
log_vars <- c('avgAnnCount', 'avgDeathsPerYear', 'popEst2015', 'studyPerCap', 'PctBachDeg18_24
log_names <- c()
all_vars <- setdiff(vars_1, log_vars)
all_vars <- append(all_vars, log_names)
```

```r
all_vars <- sort(all_vars)
df_temp <- df %>% select(all_vars)
df_temp <- na.omit(df_temp)
heatmap2 <- cor(df_temp)
heatmap.2(heatmap2, trace = 'none', margins = c(10,10), col = 'cm.colors', cexRow=0.7, cexCol =
cancer %>%
  mutate(
    high_college = pct_bach_deg25_over > 34, # median(pct_bach_deg18_24),
  ) %>%
  pivot_longer(
    cols = c("pct_white",
             "pct_black",
             "pct_asian",
             "pct_other_race"),
    values_to = "pct",
    names_to = "race"
  ) %>%
  mutate(
    race = case_when(
      race == "pct_white" ~ "White",
      race == "pct_black" ~ "Black",
      race == "pct_asian" ~ "Asian",
      race == "pct_other_race" ~ "Other"
    )
  ) %>%
  ggplot(aes(x = pct, y = target_death_rate, color = high_college)) +
  geom_smooth(method = "lm", size = 0.5) +
  facet_grid(race ~ .) +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    plot.title = element_text(hjust = 0.5)
  ) +
  labs(
    title = "Death rate by percent of by race in a county and high education",
    x = "Percentage of race in county population",
    y = "Cancer Mortality (deaths / 100K)") +
  scale_color_discrete(labels = c("Above Median", "Below Median"))
cancer_model = lm(target_death_rate ~ pct_hs25_over + pct_bach_deg25_over +
             pct_white + pct_black + pct_asian + pct_other_race +
             # Interactions
             pct_white*pct_hs25_over + pct_black*pct_hs25_over + pct_asian*pct_hs25_over + pct_
             pct_white*pct_bach_deg25_over + pct_black*pct_bach_deg25_over + pct_asian*pct_bach
             # Confounders
             incidence_rate + med_income + pop_est2015 + poverty_percent + median_age,
           data = cancer)
cancer_model %>%
```

```
  broom::tidy() %>%
  mutate(
    left = estimate - qt(0.75, df = nrow(cancer) - length(cancer_model$coefficients)) * std.er
    right = estimate + qt(0.75, df = nrow(cancer) - length(cancer_model$coefficients)) * std.e
    `95% CI` = paste0("(", left %>% round(3), ", ", right %>% round(3), ")")
  ) %>%
  filter(
    term %in% c("pct_hs25_over", "pct_bach_deg25_over",
                "pct_white", "pct_black", "pct_asian", "pct_other_race",
                "pct_hs25_over:pct_white", "pct_hs25_over:pct_black",
                "pct_hs25_over:pct_asian", "pct_hs25_over:pct_other_race",
                "pct_bach_deg25_over:pct_white", "pct_bach_deg25_over:pct_black",
                "pct_bach_deg25_over:pct_asian", "pct_bach_deg25_over:pct_other_race")
  ) %>%
  select(term, estimate, `95% CI`) %>%
  kable(digits = 3,
        caption = "Estimated coefficients and 95% CI")
bs_n = 1000
create_int_model = function(data) {
  lm(target_death_rate ~ pct_hs25_over + pct_bach_deg25_over +
               pct_white + pct_black + pct_asian + pct_other_race +
               # Interactions
               pct_white*pct_hs25_over + pct_black*pct_hs25_over + pct_asian*pct_hs25_over + pct
               pct_white*pct_bach_deg25_over + pct_black*pct_bach_deg25_over + pct_asian*pct_bac
               # Confounders
               incidence_rate + med_income + pop_est2015 + poverty_percent + median_age,
             data = data)
}
m1 = create_int_model(cancer)
terms = m1$coefficients %>% names %>% .[13:20]
# Create the bootstrap datasets and models
bs = tibble( idx = 1:bs_n ) %>%
  mutate(
    bs_data = map(idx, function(i) {
      sample_n(cancer, size = nrow(cancer), replace = TRUE)
    }),
    bs_model = map(bs_data, function(bsd) {
      create_int_model(bsd)
    }),
    bs_results = map(bs_model, broom::tidy)
  ) %>%
  select(idx, bs_results) %>%
  unnest(bs_results) %>%
  group_by(term) %>%
  summarize(
    n = n(),
    bs_mean = mean(estimate),
```

```
    bs_var = var(estimate),
    left_bound = quantile(estimate, 0.025),
    right_bound = quantile(estimate, 0.975),
  ) %>%
  filter(term %in% terms) %>%
  # Convert terms to factors for easier reordering
  mutate(
    term = factor(term,
                  levels = c(
                    "pct_hs25_over:pct_asian", "pct_bach_deg25_over:pct_asian",
                    "pct_hs25_over:pct_black", "pct_bach_deg25_over:pct_black",
                    "pct_hs25_over:pct_other_race", "pct_bach_deg25_over:pct_other_race",
                    "pct_hs25_over:pct_white", "pct_bach_deg25_over:pct_white"),
                  labels = c(
                    "HS x Asian", "College x Asian",
                    "HS x Black", "College x Black",
                    "HS x Other", "College x Other",
                    "HS x White", "College x White"
                  ))
  )
# Visualize the bootstrap confidence intervals
bs %>%
  ggplot(aes(x = term, y = bs_mean)) +
  geom_pointrange(aes(ymin = left_bound, ymax = right_bound,
                      color = if_else(left_bound > 0 | right_bound < 0, "y", "n"))
                  ) +
  geom_hline(yintercept = 0, color = "red", alpha = 0.5) +
  coord_flip() +
  labs(
    title = "Bootstrap 95% percentile intervals for 1000 resamples",
    x = "Interaction Term",
    y = "Bootstrap Interaction Coefficient Estimate"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "none"
  ) +
  scale_color_manual(values = c("#DC143C", "#2E8B57"))
dir = getwd()
data_dir <- paste(dir, '/Jasen/df_imp.RData', sep = '')
load(data_dir)
response_vars <- c('TARGET_deathRate')
df <- df_imp4 %>% select(- c('ID', 'incidenceRate'))
response_vars <- c('TARGET_deathRate')
vars <- colnames(df)
vars_1 <- setdiff(vars, response_vars)
```

```
predict_vars <- paste(vars_1, collapse = ' + ')
df <- data.frame(sapply(df, as.numeric))
set.seed(221)
test_set_index <- sample(1:nrow(df), floor(nrow(df))/10)
train_set_index <- setdiff(1:nrow(df), test_set_index)
test <- df[test_set_index,]
train <- df[train_set_index,]
test_y <- test %>% select(response_vars)
test_x <- test %>% select(vars_1)
train_y <- train %>% select(response_vars)
train_x <- train %>% select(vars_1)
lambda_seq <- 10^seq(-4, -1.95, by = .025)
set.seed(221)
cv_output <- cv.glmnet(as.matrix(train_x), as.matrix(train_y),
                       alpha = 1, lambda = lambda_seq)
best_lambda <- cv_output$lambda.min
best_lambda
lasso_best <- glmnet(as.matrix(train_x), as.matrix(train_y), alpha = 1, lambda = best_lambda)
pred_test_y <- predict(lasso_best, s = best_lambda, newx = as.matrix(test_x))
pred_train_y <- predict(lasso_best, s = best_lambda, newx = as.matrix(train_x))
cv_error <- cv_output$cvm
lambdas <- cv_output$lambda
hyper <- data.frame(lambdas, cv_error)
# taking the minimum
min_cv_error <- min(hyper$cv_error)
min_df <- hyper %>% filter(cv_error == min_cv_error)
grob <- grobTree(textGrob("(a)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
g_lasso_hyper <- ggplot() + geom_point(data = hyper, aes(x = lambdas, y = cv_error)) +
  geom_point(data = min_df, aes(x = lambdas, y = cv_error), color = 'red') +
  scale_x_continuous(trans = 'log10') +
  ylab('Mean Cross-Validation Error') +
  xlab('Lambda') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
train_results <- data.frame(train_y, pred_train_y)
colnames(train_results) <- c('y_true', 'y_hat')
R_squared <- as.numeric(unname(cor(train_y, pred_train_y)))
r_squared_lasso_train <- R_squared
R_squared <- sprintf("%.3f", round(R_squared,3))
R_squared_label <- paste('R^2 = ', R_squared)
rmse_lasso_train <- sqrt(sum((train_results$y_true - train_results$y_hat)^2)/nrow(train_results
grob <- grobTree(textGrob("(b)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
g_lasso_train <- ggplot(train_results) +
  geom_point(aes(x = y_true, y= y_hat)) +
  geom_line(aes(x = y_true, y = y_true), color = 'red')  +
```

```
  geom_label(label = R_squared_label, x = 100, y = 275, label.padding = unit(0.55, "lines"),
    label.size = 0.35,
    color = "black",
    fill="#69b3a2") +
  ylab('Predicted Cancer Death Rate') +
  xlab('True Cancer Death Rate') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
test_results <- data.frame(test_y, pred_test_y)
colnames(test_results) <- c('y_true', 'y_hat')
R_squared <- as.numeric(unname(cor(test_y, pred_test_y)))
r_squared_lasso_test <- R_squared
R_squared <- sprintf("%.3f", round(R_squared,3))
R_squared_label <- paste('R^2 = ', R_squared)
rmse_lasso_test <- sqrt(sum((test_results$y_true - test_results$y_hat)^2)/nrow(test_results))
grob <- grobTree(textGrob("(c)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
g_lasso_test <- ggplot(test_results) +
  geom_point(aes(x = y_true, y= y_hat)) +
  geom_line(aes(x = y_true, y = y_true), color = 'red')  +
  geom_label(label = R_squared_label, x = 125, y = 250, label.padding = unit(0.55, "lines"),
    label.size = 0.35,
    color = "black",
    fill="#69b3a2") +
  ylab('Predicted Cancer Death Rate') +
  xlab('True Cancer Death Rate') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
lasso_beta_hat <- as.numeric(lasso_best$beta)
lasso_var_names <- rownames(lasso_best$beta)
lasso_final_df <- data.frame(lasso_coeffs = lasso_beta_hat)
rownames(lasso_final_df) = lasso_var_names


# PCA PART ----------------------------------------------------
dir = getwd()
data_dir <- paste(dir, '/Jasen/df_imp.RData', sep = '')
load(data_dir)
response_vars <- c('TARGET_deathRate')
y <- df_imp4 %>% select(response_vars)
y <- unname(unlist(y))
do_not_include <- c('ID', 'TARGET_deathRate', 'incidenceRate', 'popEst2015_log', 'medIncome',
do_not_include <- c('ID', 'TARGET_deathRate')
df <- df_imp4 %>% select(- do_not_include)
vars_1 <- colnames(df)
df <- data.frame(sapply(df, as.numeric))
S <- cov(df)
```

```r
eig <- eigen(S)
eig_vals <- eig$values
eig_vecs <- eig$vectors
cum_var_explained <- cumsum(eig_vals/(sum(eig_vals)))
PCA_model <- prcomp(df)
PCA_loadings <- PCA_model$rotation
PCA_summary <- summary(PCA_model)
set.seed(221)
test_set_index <- sample(1:nrow(df), floor(nrow(df))/10)
train_set_index <- setdiff(1:nrow(df), test_set_index)
test <- df[test_set_index,]
train <- df[train_set_index,]
test_y <- y[test_set_index]
test_x <- test %>% select(vars_1)
train_y <- y[train_set_index]
train_x <- train %>% select(vars_1)
num_comp <- 1:ncol(train_x)
mses <- integer(ncol(train_x))
results <- data.frame()
PCA_train_model <- prcomp(train_x)
component_matrix <- data.frame(as.matrix(train_x) %*% PCA_train_model$rotation)
for(i in num_comp){

  pca_df <- data.frame(component_matrix[,1:i])


  eq <- paste(colnames(pca_df), collapse = ' + ')
  eq <- paste('deathRate', eq, sep = ' ~ ')

  pca_df <- pca_df %>% mutate(deathRate = train_y)

  # Train the model

  train.control <- trainControl(method = "cv", number = 10)

  model <- train(deathRate ~., data = pca_df, method = "lm",
                 trControl = train.control)
  if(i == 1){
    results <- model$results
  } else{
    results <- rbind(results, model$results)
  }
}
results <- results %>% mutate(ID = as.numeric(rownames(results)))
results_best <- results %>% filter(ID == 10)
grob <- grobTree(textGrob("(a)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
```

```
g_pca_hyper <- ggplot() + geom_point(data = results, aes(x = ID, y = RMSE)) +
  geom_point(data = results_best, aes(x = ID, y= RMSE), color = 'red') +
  xlab('# of PCs') +
  ylab('RMSE') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
n <- 10
train_component_matrix <- data.frame(as.matrix(train_x) %*% PCA_train_model$rotation)
pca_df <- data.frame(train_component_matrix[,1:n])
eq <- paste(colnames(pca_df), collapse = ' + ')
eq <- paste('deathRate', eq, sep = ' ~ ')
pca_df <- pca_df %>% mutate(deathRate = train_y)
pca_train_model_2 <- lm(eq, data = pca_df)
RMSE_PCA_train <- sqrt(sum(residuals(pca_train_model_2)^2)/nrow(pca_df))
y_pred <- pca_train_model_2$fitted.values
pca_df <- pca_df %>% mutate(y_pred = y_pred)
R_squared <- as.numeric(unname(cor(y_pred, train_y)))
R_squared_PCA_train <- R_squared
R_squared <- sprintf("%.3f", round(R_squared,3))
R_squared_label <- paste('R^2 = ', R_squared)
grob <- grobTree(textGrob("(b)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
g_pca_train <- ggplot(data = pca_df) +
  geom_point(aes(x = deathRate, y = y_pred)) +
  geom_line(aes(x = deathRate, y = deathRate), color = 'red') +
  geom_label(label = R_squared_label, x = 100, y = 275, label.padding = unit(0.55, "lines"),
             label.size = 0.35,
             color = "black",
             fill="#69b3a2") +
  xlab('True Cancer Death Rate') +
  ylab('Predicted Cancer Death Rate') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
n <- 10
# use the train_PCA_loadings
test_component_matrix <- data.frame(as.matrix(test_x) %*% PCA_train_model$rotation)
pca_df <- data.frame(test_component_matrix[,1:n])
eq <- paste(colnames(pca_df), collapse = ' + ')
eq <- paste('deathRate', eq, sep = ' ~ ')
### Do we fit the test set data and get beta_test_hat??
# pca_test_model <- lm(eq, data = pca_df)
# RMSE_PCA_test <- sqrt(sum(residuals(pca_test_model)^2)/nrow(pca_df))
#
# y_pred <- unname(predict(pca_test_model))
y_pred <- predict(pca_train_model_2, test_component_matrix)
RMSE_PCA_test <- sqrt(sum((y_pred - test_y)^2)/nrow(pca_df))
pca_df <- pca_df %>% mutate(y_pred = y_pred)
```

```r
pca_df <- pca_df %>% mutate(deathRate = test_y)
R_squared <- as.numeric(unname(cor(y_pred, test_y)))
R_squared_PCA_test <- R_squared
R_squared <- sprintf("%.3f", round(R_squared,3))
R_squared_label <- paste('R^2 = ', R_squared)
grob <- grobTree(textGrob("(c)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
g_pca_test <- ggplot(data = pca_df) +
  geom_point(aes(x = deathRate, y = y_pred)) +
  geom_line(aes(x = deathRate, y = deathRate), color = 'red') +
  geom_label(label = R_squared_label, x = 125, y = 250, label.padding = unit(0.55, "lines"),
             label.size = 0.35,
             color = "black",
             fill="#69b3a2") +
  xlab('True Cancer Death Rate') +
  ylab('Predicted Cancer Death Rate') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
s_PCA <- summary(pca_train_model_2)
loadings <- PCA_train_model$rotation
final_loadings <- loadings[,1:10]
# STEPWISE ------------------------------------------------------
dir = getwd()
data_dir <- paste(dir, '/Jasen/df_imp.RData', sep = '')
load(data_dir)
df <- df_imp4 %>% select(- c('ID'))
response_vars <- c('TARGET_deathRate')
vars <- colnames(df)
vars_1 <- setdiff(vars, response_vars)
predict_vars <- paste(vars_1, collapse = ' + ')
df <- data.frame(sapply(df, as.numeric))
set.seed(221)
test_set_index <- sample(1:nrow(df), floor(nrow(df))/10)
train_set_index <- setdiff(1:nrow(df), test_set_index)
test <- df[test_set_index,]
train <- df[train_set_index,]
test_y <- test %>% select(response_vars)
test_y <- unlist(unname(test_y))
test_x <- test %>% select(vars_1)
train_y <- train %>% select(response_vars)
train_y <- unlist(unname(train_y))
train_x <- train %>% select(vars_1)
step_model <- step(lm(TARGET_deathRate ~ 1, data = train), ~ incidenceRate + medIncome + povert
features <- 'PctBachDeg25_Over + incidenceRate + povertyPercent + PctHS18_24 + avgDeathsPerYea
features2 <- unlist(strsplit(features, split = ' + ', fixed = T))
train_x <- train_x %>% select(features2)
pred_train_y <- predict(step_model, data = train_x)
```

```
RMSE_step_train <- sqrt(sum((pred_train_y - train_y)^2)/length(train_y))
train_results <- data.frame(train_y, pred_train_y)
colnames(train_results) <- c('y_true', 'y_hat')
R_squared <- as.numeric(unname(cor(train_y, pred_train_y)))
R_squared_step_train <- R_squared
R_squared <- sprintf("%.3f", round(R_squared,3))
R_squared_label <- paste('R^2 = ', R_squared)
grob <- grobTree(textGrob("(b)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
g_step_train <- ggplot(train_results) +
  geom_point(aes(x = y_true, y= y_hat)) +
  geom_line(aes(x = y_true, y = y_true), color = 'red')  +
  geom_label(label = R_squared_label, x = 100, y = 275, label.padding = unit(0.55, "lines"),
    label.size = 0.35,
    color = "black",
    fill="#69b3a2") +
  ylab('Predicted Cancer Death Rate') +
  xlab('True Cancer Death Rate') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
test_x <- test_x %>% select(features2)
pred_test_y <- unname(predict(step_model, test_x))
RMSE_step_test <- sqrt(sum((pred_test_y - test_y)^2)/length(test_y))
test_results <- data.frame(test_y, pred_test_y)
colnames(test_results) <- c('y_true', 'y_hat')
R_squared <- as.numeric(unname(cor(test_y, pred_test_y)))
R_squared_step_test <- R_squared
R_squared <- sprintf("%.3f", round(R_squared,3))
R_squared_label <- paste('R^2 = ', R_squared)
grob <- grobTree(textGrob("(c)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
g_step_test <- ggplot(test_results) +
  geom_point(aes(x = y_true, y= y_hat)) +
  geom_line(aes(x = y_true, y = y_true), color = 'red')  +
  geom_label(label = R_squared_label, x = 125, y = 250, label.padding = unit(0.55, "lines"),
    label.size = 0.35,
    color = "black",
    fill="#69b3a2") +
  ylab('Predicted Cancer Death Rate') +
  xlab('True Cancer Death Rate') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
step_coeffs <- step_model$coefficients
df_step_coeffs <- data.frame(step_coeffs)
model <- lm(TARGET_deathRate ~ ., data = train)
k <- ols_step_both_aic(model)
df_AIC <- data.frame(steps = 1:k$steps, AIC = k$aic, vars = k$predictors)
```

```
grob <- grobTree(textGrob("(a)", x=0.01,  y=0.97, hjust=0,
  gp=gpar(fontsize=13)))
g_step_hyper <- ggplot(df_AIC, aes(x= steps, y= AIC, label=vars))+
  geom_point(color = 'red') +
  geom_line() +
  geom_text(aes(label=vars),hjust=0, vjust=0, angle = 45, size = 3) +
  xlim(0,30) +
  ylim(21000, 25500) +
  xlab('Steps') +
  ylab('AIC') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))
# PUTTING IT ALL TOEGHER ------------------------------------------------
final_coeffs <- merge(df_step_coeffs, lasso_final_df, by = "row.names", all = TRUE)
rownames(final_coeffs) <- final_coeffs$Row.names
final_coeffs <- final_coeffs %>% select(-c(Row.names))
#the loadings for the PCA
#final_loadings
#the betas for the 10 PC's
#summary(pca_train_model_2)
R_squared_train <- c(r_squared_lasso_train, R_squared_PCA_train, R_squared_step_train)
R_squared_test <- c(r_squared_lasso_test, R_squared_PCA_test, R_squared_step_test)
RMSE_train <- c(rmse_lasso_train, RMSE_PCA_train, RMSE_step_train)
RMSE_test <- c(rmse_lasso_test, RMSE_PCA_test, RMSE_step_test)
final_metrics <- rbind(R_squared_train, R_squared_test, RMSE_train, RMSE_test)
colnames(final_metrics) <- c('Lasso', 'PCA', 'Stepwise')
train_metrics <- rbind(R_squared_train, RMSE_train)
colnames(train_metrics) <- c('Lasso', 'PCA', 'Stepwise')
test_metrics <- rbind(R_squared_test, RMSE_test)
colnames(test_metrics) <- c('Lasso', 'PCA', 'Stepwise')
lay <- rbind(c(1,2,3),
             c(4,5,6),
             c(7,8,9))
lay1 <- rbind(c(1,2,3))
final_metrics %>% kable(digits = 3)
grid.arrange(g_lasso_hyper, g_lasso_train, g_lasso_test, layout_matrix = lay1)
grid.arrange(g_pca_hyper,g_pca_train,g_pca_test, layout_matrix = lay1)
grid.arrange(g_step_hyper,g_step_train,g_step_test, layout_matrix = lay1)
final_coeffs
```

7 Supplemental

Table Z: Comparison of Stepwise and LASSO model coefficients

```
final_coeffs
```

```
##                                   step_coeffs  lasso_coeffs
## (Intercept)                      8.982813e+02            NA
## avgAnnCount_log                 -2.755781e+00 -1.914430e+00
## avgDeathsPerYear_log             1.039750e+02  1.121735e+02
## AvgHouseholdSize                 4.653770e+00  6.639897e+00
## BirthRate                       -4.772253e-01 -6.828285e-01
## college_black                    1.823608e-02  1.490543e-02
## hs_black                                   NA -2.201629e-02
## hs_white                                   NA -1.898820e-02
## Imputed_PctEmployed16_Over      -4.832461e-01 -6.286077e-01
## Imputed_PctPrivateCoverageAlone            NA -2.531996e-02
## incidenceRate                    9.508184e-02            NA
## MedianAge                       -2.559524e+00 -2.872336e+00
## medIncome                        3.834720e-04  5.224504e-04
## PctAsian_log                               NA  3.642454e-01
## PctBachDeg18_24_log                        NA  1.415853e-02
## PctBachDeg25_Over               -1.529700e-01 -1.637155e-01
## PctBlack                        -1.887251e-01  6.371510e-01
## PctEmployed16_Over              -5.092996e-01 -6.566720e-01
## PctEmpPrivCoverage               1.053615e-01  1.436310e-01
## PctHS18_24                       3.614372e-01  4.166293e-01
## PctHS25_Over                               NA  1.865748e+00
## PctMarriedHouseholds                       NA -4.685820e-01
## PctNoHS18_24                    -6.260260e-02 -4.481827e-02
## PctOtherRace_log                -9.016052e-01 -1.265590e+00
## PctPrivateCoverage              -2.662975e-01 -5.540287e-02
## PctPrivateCoverageAlone                    NA -2.523838e-02
## PctPublicCoverage               -2.148296e+00 -2.366866e+00
## PctPublicCoverageAlone           1.644217e+00  2.046082e+00
## PctSomeCol18_24                            NA  4.212474e-02
## PctUnemployed16_Over             8.859463e-01  9.699252e-01
## PctWhite                                   NA  6.649455e-01
## PercentMarried                             NA  4.895686e-01
## popEst2015_log                  -1.010836e+02 -1.093631e+02
## povertyPercent                   3.958523e-01  5.106701e-01
## studyPerCap_log                 -4.043149e-01 -3.224617e-01
```