Final Report: Predicting Cancer Mortality On County-Level Data

Jasen Zhang, Alexander Zhu, Christian Pascual

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, affordable cancer cures and treatments represents a important front of research.

Cancer is inherently a disease of the genes, but a wide berth of research has demonstrated that a diverse set of environmental and socioeconomic factors contribute to increased risk of cancer incidence 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 of studies that have documented how race influences cancer mortality, highlighting that it is critical 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 monolithic set of symptoms. 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]. The heterogeneity of different cancers in the United States offers an interesting research avenue. Just as the aforementioned studies examined how socioeconomic factors and race affect cancer mortality on an individual, it could be useful to understand how these factors relate to cancer mortality on a higher, geographic level.

Understanding how these factors contribute to cancer mortality on a geographic level offers an opportunity for researchers to understand the factors that contribute to higher mortality in different states and possibly a better way to allocate health resources to areas that are harder hit.

Data

For our analysis, we will use a dataset aggregated from multiple sources, which we note later. The data spans from 2010 to 2016 and includes information on 3047 counties in the United States. There are a total of 3,143 counties and county-equivalents in the United States, so 3.1% of them are represented in the data. Our group did not aggregate the data ourselves, it is publically available and can be found here.

Our data contains information on various demographic, socioeconomic, household, and cancer-related factors for each county, represented typically as percentages. These data are gathered from the 2013 Census. Each row contains the percentage of each race (Asian, Black, White and Other) that live in the county. The data also contains information on educational acheivement as well, measured as the proportion of the county population who have achieved high school and college degrees. The proportion of people who have public and private health insurance is another notable variable in the data.

The cancer data has been aggregated from the American Community Survey, cancer.gov and clinicaltrials.gov, spanning from 2010 to 2016. In terms of important cancer-related factors, we have 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.

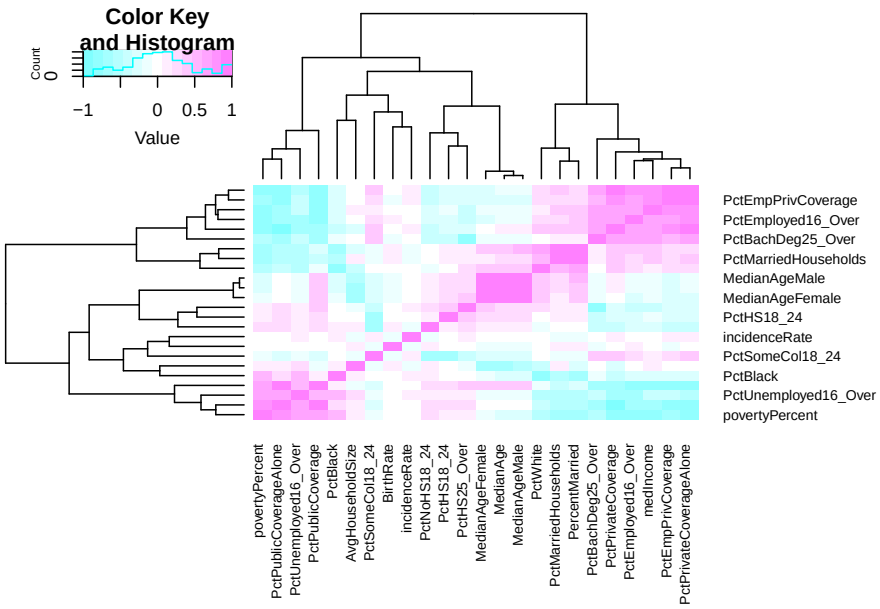
Our target response is cancer mortality, also measured as mean per capita. In this dataset, cancer mortality ranges from 59.7 to 362.8, with a median value of 178.1. The outcome is also reasonably bell-shaped, so we don't think any transformation will be necessary for the analyses.

Initial Analyses & Objectives

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

Correlation Between Predictors

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



The `povertyPercent` variable correlates highly with other predictors that deal with being under public insurance of being unemployed. Conversely, poverty is highly negatively correlated with having private insurance. The median ages of men and women are highly correlated as well, so we will probably opt for just one of these. Since the educational variable represent highest achievement, many of them are correlated as well.

In making our future models, we now know that we'll have to deal with this correlation and perform some model selection on our variables. We performed an initial PCA to figure out what variables explained most of the variance in average cancer mortality, and this will help guide our final model selection.

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).

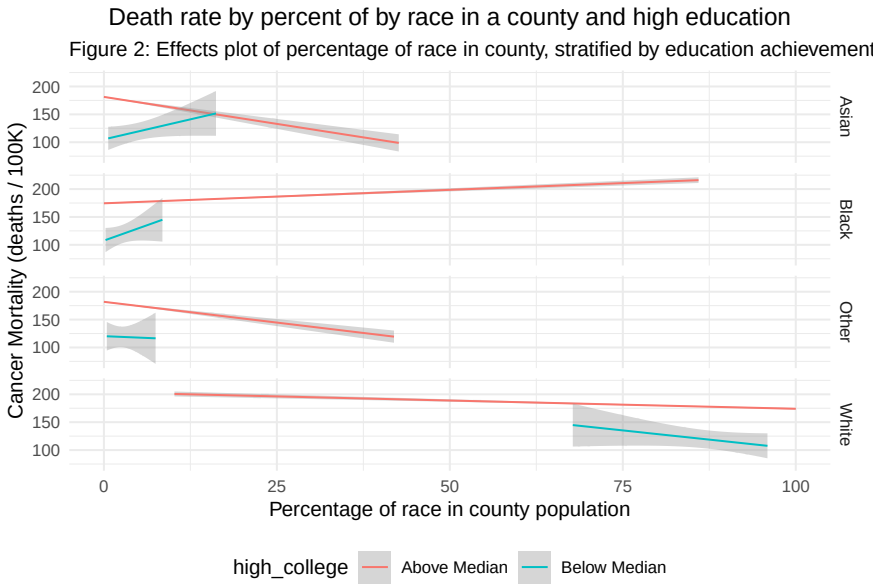


Figure 2: Effects plot of percentage of race in county, stratified by education achievement

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.

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, might they be helpful in increasing predictive ability?

Our two questions is motivated by the nature of the data. The data is contains most of the counties in the United States, so any estimates we find apply directly to these counties during this time period. Since the data is a "snapshot" of the population from 2011 - 2016, we must acknowledge that the estimates we get in any explanatory model will reflect population characteristics. We also acknowledge that some variation is introduced because the data is come different points in time, we assume that these measurements do not change drastically on the scale of a few years. With the nature of the data in mind, we want to try to use the explanatory model to see if any features or combination of features might be useful in a prediction model, based on the estimated coefficients.

Methodology

Explanatory Model

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.

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.

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.

Prediction Model

We proceeded to build 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, principal component analysis (PCA), and forward backwards stepwise selection model. We randomly split the data into 80% train and 20% test, and then used this same split for all three models. Afterwards, for the lasso and PCA models, the training set was further split into 10 folds for cross-validation in order to tune the lambda and number of PC hyperparameters respectively. Choosing 10 folds and an $80 - 20$ split are usually default parameters in machine learning models.

Lasso

In lasso regularization, the aim is to minimize not just the residual sum of squares (RSS), but the sum of RSS and a scalar multiple of the absolute sum of the beta coefficients. This allows for the moderation of beta coefficients and serves to filter out features that do minimal explanation. We performed 10 fold cross-validation to select for the optimal value of λ, the scalar multiple, and then fitted the lasso regularization model on the entire training set. The beta estimates were then used to predict cancer death rates in the test set.

PCA

In PCA, we used 10 fold cross validation to select for an optimal number of principal components (PCs). Since the error formula we used was root mean square error (RMSE), adding additional PCs generally decrease RMSE. As a result, we used a heuristic, the elbow rule, to select for the optimal amount of PCs. The elbow rule selects an optimal value for the number of PCs when adding additional PCs no longer significantly reduces RMSE. Once we selected the optimal number

of PCs, k, we computed the first k PC loadings and used those loadings to predict the cancer death rates in the train and test sets.

Forward and Backward Stepwise Selection

Forward and backwards stepwise selection did not use cross-validation to select for a hyperparmeter. Instead, we used the stepwise selection algorithm with AIC as a metric on the entire training set to select for a set of variables. These variables were then fitted on the training set and used to predict the cancer death rates in the test set.

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. The estimated beta coefficients in the lasso and stepwise models and the loadings in the PCA model can be used to interpret prediction results.

Results & Discussion

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. This lines up with previous research suggesting that education is beneficial for health outcomes.

Table 1: 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)
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)

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, hich matches with our original estimated model.

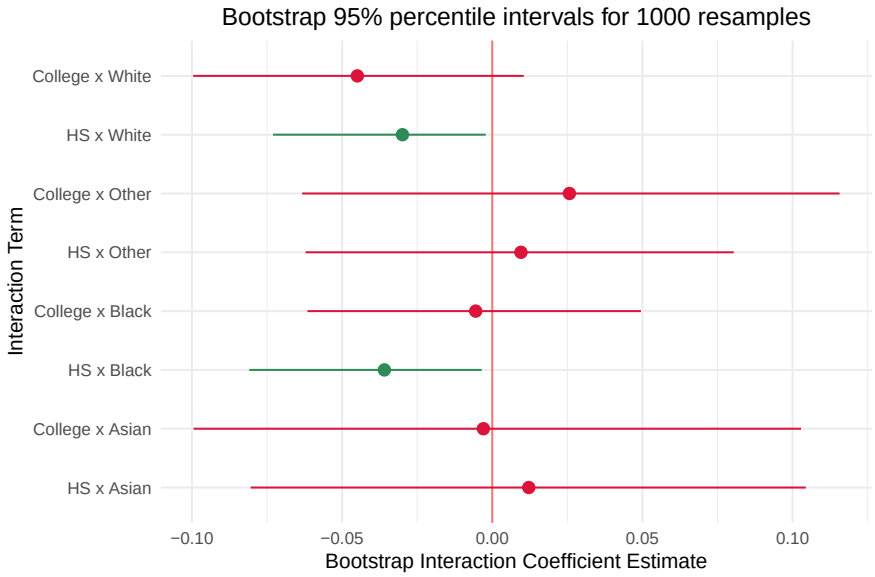


Figure 3

With these findings, we include these two interaction terms inside of a candidate predictive model.

Prediction Model

The lasso regularization model selected an optimal hyperparameter of $\lambda = 0.00119$ as shown in Figure X (a). The estimated coefficients fitted from the training set are shown in Table Y, with only the intercept and incidenceRate having coefficients of 0. The training set RMSE and R^2 was 13.796 and 0.866 respectively, and the test set metrics showed slight improvements with an RMSE and R^2 of 12.842 and 0.872 respectively. These values are in Table Z.

Using the elbow rule as shown in Figure C (a), we selected 10 PCs to be the optimal amount and obtained the loadings of the first 10 PCs from the training set. A summary of these loadings are in Table A, where roughly half of the variables did not show up significantly in any of the 10 PCs. The training set RMSE and R^2 was 20.020 and 0.689 respectively, and the test set metrics showed moderate improvements with an RMSE and R^2 of 17.955 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 13.278 and 0.877 respectively, and the test set metrics showed slight improvements with an RMSE and R^2 of 12.495 and 0.879 respectively. These values are in Table Z.

```
final_metrics
```

```
##                      Lasso        PCA   Stepwise
## R_squared_train  0.8664342  0.6892268  0.8769632
## R_squared_test   0.8718580  0.7281111  0.8794552
```

```
## RMSE_train      13.7961626 20.0200150 13.2783463
## RMSE_test       12.8424256 17.9551337 12.4945712
```

Table Z

```
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
```

Table Y

```
grid.arrange(g_lasso_hyper,g_lasso_train,g_lasso_test, layout_matrix = lay1)
```

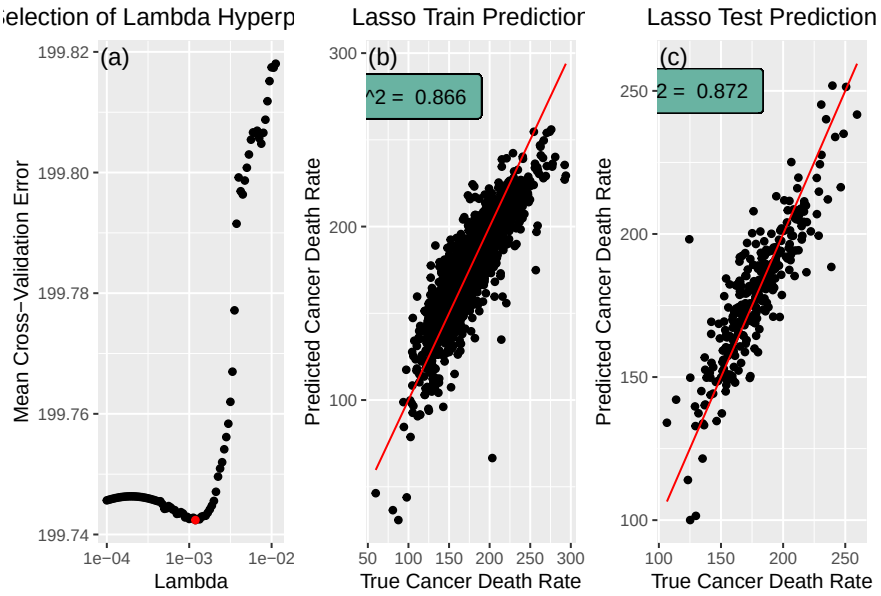


Figure X

```
grid.arrange(g_pca_hyper,g_pca_train,g_pca_test, layout_matrix = lay1)
```

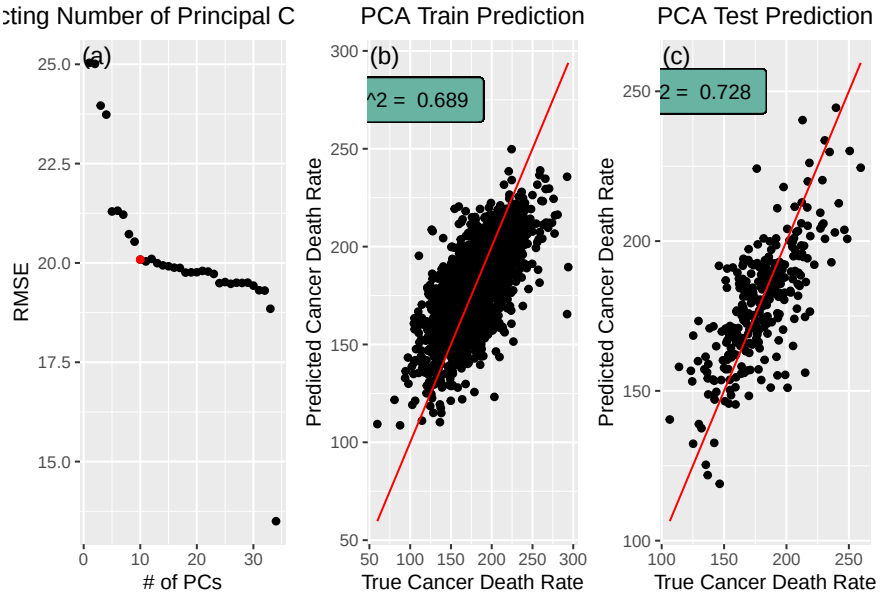


Figure C

```
grid.arrange(g_step_hyper,g_step_train,g_step_test, layout_matrix = lay1)
```

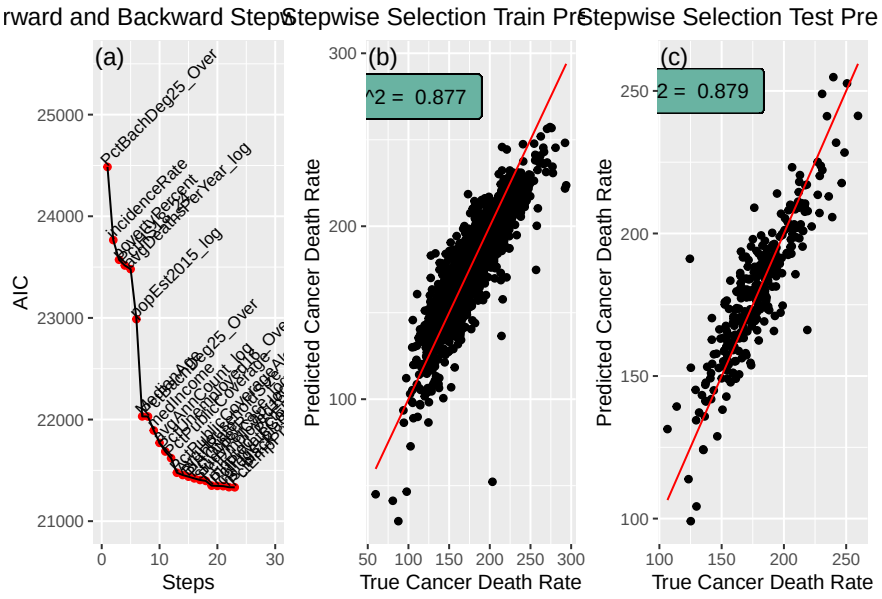


Figure B

```
final_loadings
```

```
##                                        PC1           PC2           PC3
## incidenceRate                -4.906072e-05 -3.458332e-03 -3.015126e-02
## medIncome                    -9.997924e-01 -9.091488e-03 -1.808253e-02
## povertyPercent                4.154681e-04  2.494200e-03 -6.532604e-04
## MedianAge                     5.354222e-05 -2.276568e-03 -1.633544e-03
## AvgHouseholdSize             -3.305854e-06  7.998591e-05  7.136982e-05
## PercentMarried               -1.940984e-04 -4.366232e-03  2.376839e-03
## PctNoHS18_24                  1.943937e-04  3.499056e-04 -1.201002e-03
## PctHS18_24                    1.413698e-04 -2.459173e-03 -4.519889e-03
## PctSomeCol18_24              -1.525707e-04  1.671578e-03  5.033931e-03
## PctHS25_Over                  2.795490e-04 -4.569666e-03 -9.369580e-03
## PctBachDeg25_Over            -3.140969e-04  1.558766e-03  3.850017e-03
## PctEmployed16_Over           -4.460714e-04 -8.560957e-04  2.613779e-03
## PctUnemployed16_Over          1.257925e-04  1.369913e-03 -9.348405e-04
## PctPrivateCoverage           -6.329772e-04 -3.100880e-03 -1.845921e-03
## PctPrivateCoverageAlone      -5.243248e-04 -1.921167e-03 -2.888842e-03
## PctEmpPrivCoverage           -5.784454e-04 -1.658185e-03 -3.167077e-03
## PctPublicCoverage             4.895938e-04 -2.904349e-04 -8.102522e-04
## PctPublicCoverageAlone        3.625061e-04  1.211969e-03 -1.343620e-04
## PctWhite                     -2.063774e-04 -1.487998e-02  1.068190e-02
## PctBlack                      3.099063e-04  1.125577e-02 -2.284246e-02
## PctMarriedHouseholds         -2.352540e-04 -3.643326e-03  1.530868e-03
## BirthRate                     1.074871e-06 -4.212607e-05  4.317079e-04
## Imputed_PctEmployed16_Over   -2.847074e-05 -1.674489e-04  2.589367e-05
## Imputed_PctPrivateCoverageAlone -1.284187e-04  3.555906e-04  1.339068e-03
## avgAnnCount_log              -4.106676e-05  2.866863e-04  1.968440e-04
## avgDeathsPerYear_log         -2.932705e-05  3.535126e-04 -5.207280e-06
```

```
## popEst2015_log                  -4.137450e-05  4.698798e-04  1.573582e-04
## studyPerCap_log                 -5.279787e-05  5.119546e-04  6.458466e-04
## PctBachDeg18_24_log             -3.946729e-05 -7.634365e-06 -2.046766e-06
## PctAsian_log                    -5.319556e-05  4.846992e-04  7.354057e-04
## PctOtherRace_log                -2.661135e-05  6.287687e-04  1.066399e-03
## hs_black                         1.324916e-02  3.712419e-01 -8.961406e-01
## hs_white                         1.538265e-02 -9.184454e-01 -3.920328e-01
## college_black                    8.043098e-04  1.345255e-01 -2.029164e-01
##                                           PC4           PC5           PC6
## incidenceRate                   -0.1255511821 -0.9900370682 -2.777466e-03
## medIncome                        0.0018017033  0.0004180523  3.653353e-04
## povertyPercent                   0.0028421778 -0.0022597616  2.017156e-03
## MedianAge                        0.0039004749  0.0035773653  6.917757e-03
## AvgHouseholdSize                 0.0002758510  0.0005120483 -6.476944e-05
## PercentMarried                   0.0076749257  0.0123081795  2.583746e-03
## PctNoHS18_24                     0.0187182942  0.0238726600  1.143759e-02
## PctHS18_24                       0.0124774176  0.0022677496  8.203723e-03
## PctSomeCol18_24                 -0.0215893978 -0.0236167538 -1.974094e-02
## PctHS25_Over                     0.0069497850 -0.0004548677 -1.854448e-03
## PctBachDeg25_Over               -0.0112663844  0.0003074223 -4.326691e-04
## PctEmployed16_Over              -0.0096115494 -0.0020518776 -5.214565e-02
## PctUnemployed16_Over            -0.0009041992 -0.0062921095  3.561810e-03
## PctPrivateCoverage              -0.0133579997 -0.0123139952 -1.821871e-02
## PctPrivateCoverageAlone         -0.0258428205 -0.0002829178 -7.158744e-01
## PctEmpPrivCoverage              -0.0115737697 -0.0194228541 -2.090703e-02
## PctPublicCoverage                0.0061966160 -0.0096855947  1.467622e-02
## PctPublicCoverageAlone           0.0038969118 -0.0085416665  8.455444e-03
## PctWhite                         0.0045601310 -0.0008443664  3.706167e-03
## PctBlack                        -0.0181169256  0.0019457048 -3.278570e-04
## PctMarriedHouseholds             0.0115806759  0.0183592422  3.111408e-03
## BirthRate                        0.0016154594  0.0034117913 -1.556177e-03
## Imputed_PctEmployed16_Over       0.0004241153  0.0004273098  3.811283e-02
## Imputed_PctPrivateCoverageAlone  0.0127661961 -0.0118908930  6.933157e-01
## avgAnnCount_log                 -0.0043110555 -0.0076358730 -3.195946e-04
## avgDeathsPerYear_log            -0.0047189107 -0.0055666361  5.632209e-04
## popEst2015_log                  -0.0046594854 -0.0044140687  1.964825e-04
## studyPerCap_log                 -0.0061135801 -0.0081772362 -1.601379e-03
## PctBachDeg18_24_log             -0.0025683251 -0.0027085191 -2.762398e-04
## PctAsian_log                    -0.0031996503 -0.0037813205  5.982809e-04
## PctOtherRace_log                 0.0004946635  0.0029924298  1.575223e-04
## hs_black                         0.2407662093 -0.0049660609 -5.150274e-03
## hs_white                        -0.0431873798  0.0205089081  4.010692e-03
## college_black                   -0.9598320345  0.1289877930  2.944341e-02
##                                           PC7           PC8           PC9
## incidenceRate                    8.684042e-03 -0.0362244361  0.010503239
## medIncome                        4.215921e-04 -0.0006410941 -0.001062094
## povertyPercent                   2.347184e-02 -0.0493721909 -0.175035031
## MedianAge                        1.086476e-02 -0.0579780527  0.040064365
```

```
## AvgHouseholdSize                 1.281368e-03 -0.0042882614 -0.005825779
## PercentMarried                  -1.232634e-02 -0.0352419083  0.222312821
## PctNoHS18_24                     5.348475e-02 -0.2978725600 -0.018240160
## PctHS18_24                       5.235641e-02 -0.4090464090  0.378464123
## PctSomeCol18_24                 -8.405795e-02  0.6470790612 -0.385483708
## PctHS25_Over                     7.995668e-03 -0.0670639649 -0.068593215
## PctBachDeg25_Over               -1.542547e-02  0.0898985113  0.098110418
## PctEmployed16_Over              -7.384286e-01 -0.0487841914  0.080810802
## PctUnemployed16_Over             2.556550e-02 -0.0486391980 -0.136751994
## PctPrivateCoverage              -7.135685e-02  0.2971863081  0.430311919
## PctPrivateCoverageAlone          1.337967e-02  0.1210674635  0.171786181
## PctEmpPrivCoverage              -5.939348e-02  0.2274566552  0.220673500
## PctPublicCoverage                5.032986e-02 -0.1523415894 -0.296553295
## PctPublicCoverageAlone           3.935570e-02 -0.1377931042 -0.301524414
## PctWhite                        -2.445263e-02  0.1853862365  0.233797227
## PctBlack                         2.596978e-04 -0.0022852438 -0.009117773
## PctMarriedHouseholds             8.846414e-03 -0.0382077150  0.155803426
## BirthRate                       -5.992504e-04 -0.0123443431  0.012765385
## Imputed_PctEmployed16_Over       6.471937e-01  0.1949936729  0.168645045
## Imputed_PctPrivateCoverageAlone -8.700357e-02  0.1597378723  0.184952549
## avgAnnCount_log                 -2.049905e-03  0.0137421327 -0.013572889
## avgDeathsPerYear_log             1.312474e-03  0.0070592335 -0.028896743
## popEst2015_log                   1.136788e-03  0.0090853176 -0.028487186
## studyPerCap_log                 -5.118682e-03  0.0312387434 -0.006062260
## PctBachDeg18_24_log             -7.436677e-03  0.0215244811  0.009266532
## PctAsian_log                    -6.034725e-05  0.0140037090 -0.014666663
## PctOtherRace_log                 1.213401e-04 -0.0109537158 -0.011731918
## hs_black                        -5.850860e-03  0.0154101544  0.002850570
## hs_white                         5.850028e-05  0.0004046477 -0.008784993
## college_black                    9.630910e-03 -0.0282457621  0.001764177
##                                          PC10
## incidenceRate                   -2.860960e-02
## medIncome                        2.450144e-05
## povertyPercent                   7.610751e-02
## MedianAge                       -2.193138e-01
## AvgHouseholdSize                 6.782392e-03
## PercentMarried                  -2.738575e-01
## PctNoHS18_24                    -5.794527e-01
## PctHS18_24                       5.212191e-01
## PctSomeCol18_24                  7.143079e-02
## PctHS25_Over                     1.300351e-01
## PctBachDeg25_Over               -4.895124e-02
## PctEmployed16_Over               6.896812e-03
## PctUnemployed16_Over             6.033483e-02
## PctPrivateCoverage              -5.500194e-02
## PctPrivateCoverageAlone          3.492956e-02
## PctEmpPrivCoverage               1.929601e-01
## PctPublicCoverage               -8.237150e-02
```

```
## PctPublicCoverageAlone           6.340484e-02
## PctWhite                        -3.661139e-01
## PctBlack                        -8.113672e-03
## PctMarriedHouseholds            -1.967987e-01
## BirthRate                       -4.485164e-03
## Imputed_PctEmployed16_Over       2.365081e-02
## Imputed_PctPrivateCoverageAlone  5.224743e-02
## avgAnnCount_log                  1.821538e-02
## avgDeathsPerYear_log             3.292264e-02
## popEst2015_log                   3.977256e-02
## studyPerCap_log                  3.521042e-02
## PctBachDeg18_24_log              1.576685e-02
## PctAsian_log                     3.704192e-02
## PctOtherRace_log                 1.549598e-02
## hs_black                        -5.747527e-03
## hs_white                         2.499088e-03
## college_black                   -1.226432e-02
```

Table A

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. Although this effect was small, their significance was supported simultaneously by the model itself, the Wald test, and the 95% bootstrap percentile intervals. These results support the hypothesis that higher education has a beneficial effect on cancer mortatlity, at least on the county scale. Furthermore, our findings are in line with a wealth of literature that suggest that education is beneficial to improving health outcomes.

The stepwise selection model showed the best RMSE and R^2 in the training set. Therefore, we select the stepwise selection model to predict cancer death rates for the test set. We also see that the stepwise selection model had the best metrics for the test set. The stepwise selection model also included less beta coefficients (24) than the lasso regularization model (33), so it's much more parsimonious than lasso which had marginally worse prediction metrics. Most variables in the stepwise selection model had statistically significant betas which are indicators of cancer death rates.

These findings are limited by the fact that the observational unit of the data was a county. The model suggests a beneficial benefit, but does not elucidate any reason as to why or how education interacts with race to reduce cancer mortality among African- and white Americans. The educational and race data come from 2013, so the model may be limited in its generalizability to future years, especially in the context of changing demographics and new educational environments created by COVID-19. In light of these weaknesses, we feel that our model offers an interesting perspective on how different demographic factors interact to affect a health-related outcome. Being so prevalent, it's important to understand these factors and interactions so that we can design proper interventions.

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

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 = "70%"
)

# 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(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)
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",
    subtitle = "Figure 2: Effects plot of percentage of race in county, stratified by educatio
    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_bac
             # 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
```

```r
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",
    caption = "Figure 3"
  ) +
  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)
```

```r
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') +
  ggtitle('Selection of Lambda Hyperparameter') +
  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') +
  ggtitle('Lasso Train Prediction') +
  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') +
  ggtitle('Lasso Test Prediction') +
  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
```

```r
# 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)
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)
```

```r
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') +
  ggtitle('Selecting Number of Principal Components') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))


n <- 10
```

```r
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') +
  ggtitle('PCA Train Prediction') +
  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 = ' ~ ')
```

```r
### 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)
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') +
  ggtitle('PCA Test Prediction') +
  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') +
  ggtitle('Stepwise Selection Train Prediction') +
  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') +
  ggtitle('Stepwise Selection Test Prediction') +
  annotation_custom(grob) +
  theme(plot.title = element_text(hjust = 0.5))



step_coeffs <- step_model$coefficients
df_step_coeffs <- data.frame(step_coeffs)
```

```r
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') +
  ggtitle('Forward and Backward Stepwise Model') +
  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
final_coeffs
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_loadings
```