Jayoung Kang

Minilab 3

2. Compute descriptive (summary) statistics for the subset of Opportunity Insights and PM COVID variables you filtered in previous question.

	intersects_msa	cur_smoke_q1	cur_smoke_q2	cur_smoke_q3	cur_smoke_q4
count	3107.000000	3107.000000	3107.000000	3107.000000	3107.000000
mean	0.596717	0.212659	0.171048	0.134467	0.098316
std	0.490636	0.149348	0.128130	0.132181	0.110110
min	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000	0.000000
50%	1.000000	0.250000	0.198718	0.142857	0.096535
75%	1.000000	0.310931	0.250000	0.200000	0.148719
max	1.000000	1.000000	1.000000	1.000000	1.000000

	bmi_obese_q1	bmi_obese_q2	bmi_obese_q3	bmi_obese_q4	\
count	3107.000000	3107.000000	3107.000000	3107.000000	
mean	0.239166	0.214580	0.209621	0.186739	
std	0.165928	0.153237	0.175849	0.167227	
min	0.000000	0.000000	0.000000	0.000000	
25%	0.080128	0.000000	0.000000	0.000000	
50%	0.272076	0.241590	0.223124	0.194118	
75%	0.335532	0.304348	0.297220	0.266667	
max	1.000000	1.000000	1.000000	1.000000	

	exercise_any_q1	exercise_any_q2	exercise_any_q3	exercise_any_q4
count	3107.000000	3107.000000	3107.000000	3107.000000
mean	0.455995	0.555671	0.603792	0.638727
std	0.273874	0.322336	0.357861	0.376922
min	0.000000	0.000000	0.000000	0.000000
25%	0.312500	0.444444	0.354167	0.400000
50%	0.566563	0.707143	0.778364	0.833333
75%	0.641509	0.769231	0.841804	0.890497
max	1.000000	1.000000	1.000000	1.000000

	brfss_mia	puninsured2010	reimb_penroll_adj10	mort_30day_hosp_z
count	3107.000000	3107.000000	3103.000000	3106.000000
mean	0.249437	18.469460	9302.737743	0.457806
std	0.432757	5.536651	1590.926253	1.206493
min	0.000000	3.625483	3663.530000	-7.778000
25%	0.000000	14.410247	8159.340000	-0.255867
50%	0.000000	18.147072	9193.770000	0.400088
75%	0.000000	21.961417	10285.430000	1.147822
max	1.000000	41.366287	18443.220000	8.472745

	adjmortmeas_amiall30day	adjmortmeas_chfall30day	med_prev_qual_z
count	3106.000000	3107.000000	3012.000000
mean	0.165483	0.108969	-0.148547
std	0.039408	0.023565	0.863881
min	0.000000	0.000000	-4.853847
25%	0.145312	0.096301	-0.615591
50%	0.162727	0.107242	-0.090228
75%	0.183402	0.120155	0.444429
max	0.444663	0.344451	3.478521

	primcarevis_10	diab_hemotest_10	diab_eyeexam_10	diab_lipids_10
count	3098.000000	3069.000000	3054.000000	3057.000000
mean	80.865348	83.706025	66.080221	78.307420
std	7.401457	6.594153	7.598549	7.854145
min	18.331749	16.911765	31.372549	19.661336
25%	78.803466	81.108462	61.258165	75.000000
50%	82.202779	84.782609	65.977073	79.759036
75%	84.957865	87.679083	70.906526	83.342526
max	95.665079	100.000000	90.000000	94.482759

	mammogram_10	cs00_seg_inc	cs00_seg_inc_pov25	cs00_seg_inc_aff75
count	3029.000000	3107.000000	3107.000000	3107.000000
mean	63.110073	0.025892	0.024278	0.026463
std	8.397699	0.030576	0.030757	0.032920
min	30.000000	-0.013363	-0.019502	-0.001993
25%	57.943925	0.005047	0.004164	0.003455
50%	63.618290	0.013647	0.013136	0.012577
75%	68.907563	0.036453	0.034737	0.037337
max	95.238095	0.438241	0.749106	0.196959

	cs_race_theil_2000	gini99	poor_share	inc_share_1perc	\
count	3107.000000	3008.000000	3107.000000	3008.000000	
mean	0.075402	0.379021	0.141739	0.094808	
std	0.084131	0.086677	0.065460	0.050631	
min	0.000000	0.160954	0.000000	0.018570	
25%	0.015591	0.317518	0.095383	0.062577	
50%	0.047192	0.369998	0.129621	0.083600	
75%	0.104508	0.429472	0.175282	0.113570	
max	0.712014	1.091437	0.569170	0.734770	

	frac_middleclass	scap_ski90pcm	rel_tot	cs_frac_black
count	3106.000000	3107.000000	3106.000000	3107.000000
mean	0.554244	0.000182	53.224564	8.744503
std	0.093099	1.347960	18.502524	14.483719
min	0.215630	-4.258739	1.816347	0.000000
25%	0.491883	-0.964225	39.669796	0.264501
50%	0.559830	-0.091105	51.328668	1.691121
75%	0.622758	0.818039	64.786780	10.031043
max	0.875000	9.911112	164.527310	85.965088

	cs_frac_hisp	unemp_rate	cs_labforce	cs_elf_ind_man	\
count	3107.000000	3107.000000	3107.000000	3107.000000	
mean	6.209190	0.049871	0.609344	0.159118	
std	12.050404	0.017738	0.070393	0.090862	
min	0.082034	0.016092	0.319209	0.000000	
25%	0.917235	0.037422	0.567037	0.088637	
50%	1.783438	0.046908	0.616551	0.149391	
75%	5.107685	0.058742	0.657982	0.219933	
max	97.539047	0.176995	0.860937	0.485540	

	cs_born_foreign	mig_inflow	mig_outflow	pop_density	\
count	3107.000000	3017.000000	3017.000000	3107.000000	
mean	3.441958	0.028677	0.027522	244.325026	
std	4.836270	0.019034	0.013780	1676.096088	
min	0.000000	0.000000	0.000000	0.099542	
25%	0.898505	0.016502	0.018767	17.479568	
50%	1.727323	0.024430	0.025111	43.130142	
75%	3.922074	0.036320	0.033038	104.991115	
max	50.935669	0.168671	0.153256	66940.078000	

	frac_traveltime_lt15	hhinc00	median_house_value	ccd_exp_tot
count	3107.000000	3107.000000	3.107000e+03	3080.000000
mean	0.403803	32853.502978	1.121801e+05	6.092697
std	0.137215	6975.837500	6.318905e+04	2.103573
min	0.099878	10511.805000	0.000000e+00	3.032457
25%	0.299927	28733.524500	7.704740e+04	5.027049
50%	0.385816	32234.641000	1.007748e+05	5.785282
75%	0.499088	36039.471000	1.285012e+05	6.735288
max	0.817636	77942.648000	1.333001e+06	53.258174

	score_r	cs_fam_wkidsinglemom	subcty_exp_pc	taxrate	\
count	3069.000000	3107.000000	3107.000000	3107.000000	
mean	0.077348	0.194598	2119.407531	0.023089	
std	9.007980	0.067828	999.833466	0.013848	
min	-38.687138	0.024793	0.000000	0.000000	
25%	-4.969633	0.152436	1510.192750	0.014993	
50%	0.834938	0.182469	1935.919400	0.020339	
75%	5.990181	0.221578	2505.411100	0.027164	
max	32.985218	0.543878	20541.918000	0.209907	

	tax_st_diff_top20	summer_tmmx	summer_rmax	winter_tmmx	winter_rmax
count	3106.000000	3107.000000	3107.000000	3107.000000	3107.000000
mean	0.775634	303.126997	88.970517	280.404875	87.469432
std	1.470989	3.173950	9.689271	6.597855	4.811207
min	0.000000	290.455540	31.643282	264.693820	58.159798
25%	0.000000	300.848035	88.052494	275.113020	85.093342
50%	0.000000	303.290440	91.320313	280.154690	88.028793
75%	1.000000	305.817430	94.812389	285.543750	90.747704
max	7.220000	313.872680	99.778748	298.340360	97.672874

	pm25	bmcruderate	pm25_mia	deathspc
count	3107.000000	3107.00000	3107.000000	3107.000000
mean	8.371871	1029.15597	0.003540	23.790131
std	2.565927	248.38181	0.059405	67.852145
min	0.000000	189.30000	0.000000	0.000000
25%	6.309710	864.29999	0.000000	0.000000
50%	8.784647	1036.30000	0.000000	3.802303
75%	10.483764	1194.10000	0.000000	21.461759
max	15.786018	1978.60000	1.000000	2279.610600

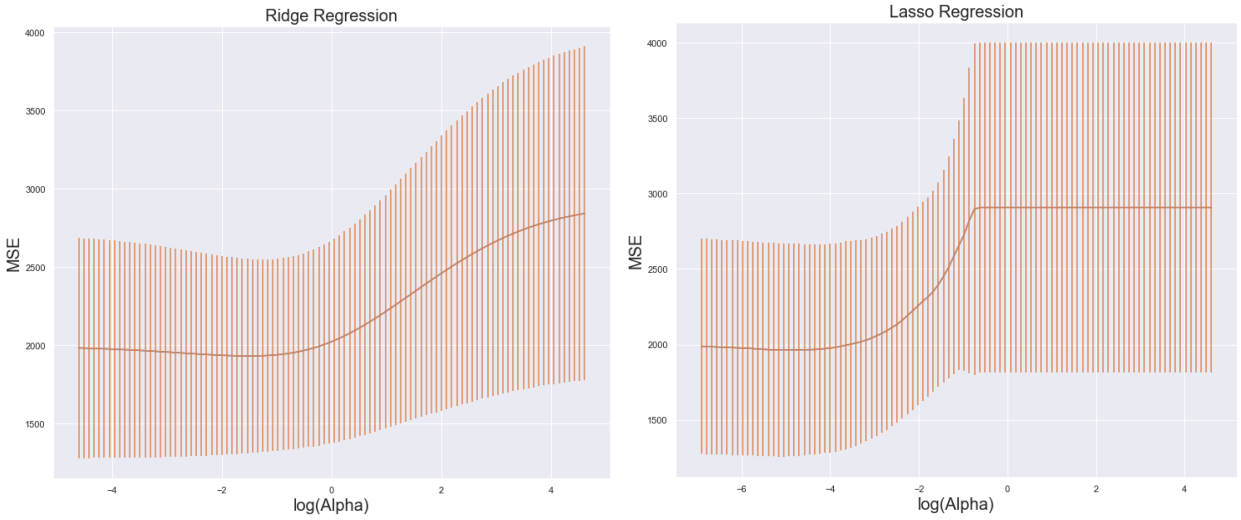
3. Note that some variables have missing values. This causes problems when estimating the models. Normally we'd impute missing values by replacing them with their mean or median value, but to keep things simple, given the size of our data, you should drop all observations (rows) with missing values.
   - Number of rows with missing variables: 92 rows

4. Create a separate dummy variable for each of the 48 states and the District of Columbia in the dataset (so you'll create 49 dummy variables in total).
   - Due to dropping the rows with NAs I the previous question there are only 47 states to make as dummy variables
   - Dropped states: DC and New Jersey

6. Using the training data, estimate the relationship between COVID-19 deaths per capita (y = deathspc) and the Opportunity Insights and PM COVID predictors listed in the spreadsheet, as well as state-level fixed effects (the state dummy variables) using OLS
   a. Based on those estimates, calculate and report the MSE and R2 in both the training and test sets.
      - Training set R-squared: 0.4129
      - Test set R-squared: 0.3798
      - Training set MSE: 1706.2683
      - Test set MSE: 1235.1068
      - There is no evidence of overfitting because the training set MSE is higher than the test set MSE. We would expect the test MSE to be much higher than the training

MSE if there were overfitting because it means the model did not generalize well and the variance is high. OLS will be prone to overfitting if we have wide data where p > n. However in this case, n > p and due to the variation in sampling we are observing a case where training MSE is higher than test MSE, as well as the R-squared.

7. Use the training set to estimate Ridge Regression and the Lasso analogs to the OLS model in the previous question.
   - Optimal alpha for the Ridge Regression: 0.2154
   - Optimal alpha for the Lasso: 0.0064



8. Using the optimal values of λ you found for Ridge Regression and the Lasso in the previous question, calculate and report the training- and test-set prediction errors (MSE & R2) for each model. Did Ridge Regression and/or the Lasso mitigate overfitting? Briefly explain your results.
   - Ridge Regression
     - training set R-squared: 0.3909
     - test set R-squared: 0.3812
     - training set MSE: 1769.9913
     - test set MSE: 1232.3872
   - Lasso
     - training set R-squared: 0.4021
     - test set R-squared: 0.3945
     - training set MSE: 1737.5940
     - test set MSE: 1206.0038
   - For both the Ridge Regression and the Lasso, the MSE and R-squared are lower for the test set than the training set. Since the OLS was already doing a pretty good job of not overfitting, the RR and Lasso didn't necessarily mitigate overfitting. However, if we had the case where n < p the RR and Lasso would have been helpful for reducing overfitting.

9. Bonus question: now compare the test-set prediction errors from Ridge Regression and the Lasso to that from OLS. Is this what you expected? Briefly explain.

- The test set MSE is lower and the test set R-squared is higher for the Lasso and the RR when compared to the OLS. This is expected because regularization adds prior constraints to the model to prevent overfitting. Regularization usually decreases in-sample fit to improve out-of-sample fit and this is what we observe for the Lasso and RR. The training set MSE is higher and the R-squared is lower for the Lasso and the RR when compared to the OLS, showing that they increased bias to lower variance based on the bias-variance tradeoff.
- In addition, when comparing the Lasso and RR, we can see the prediction errors the Lasso did better. The Lasso usually performs better than the RR when the outcome is a function of relatively few predictors. However this was difficult to know for certain a priori for the COVID-19 deaths per capita that we are dealing with. I would have actually expected 'deathspc' to be a function of relatively many predictors because there could be a wide range of factors that are associated with the Covid deaths.