

Relationship investigation between demographic variables and death rate

Analysis:

Determining the distribution of the variable of interest (death rate per 100,000 population) with the help of histogram plot. Distribution in this plot looks like normal and we are going use normal distribution for our analysis.

See appendix 1) and 2) for corresponding R-script and Plot.

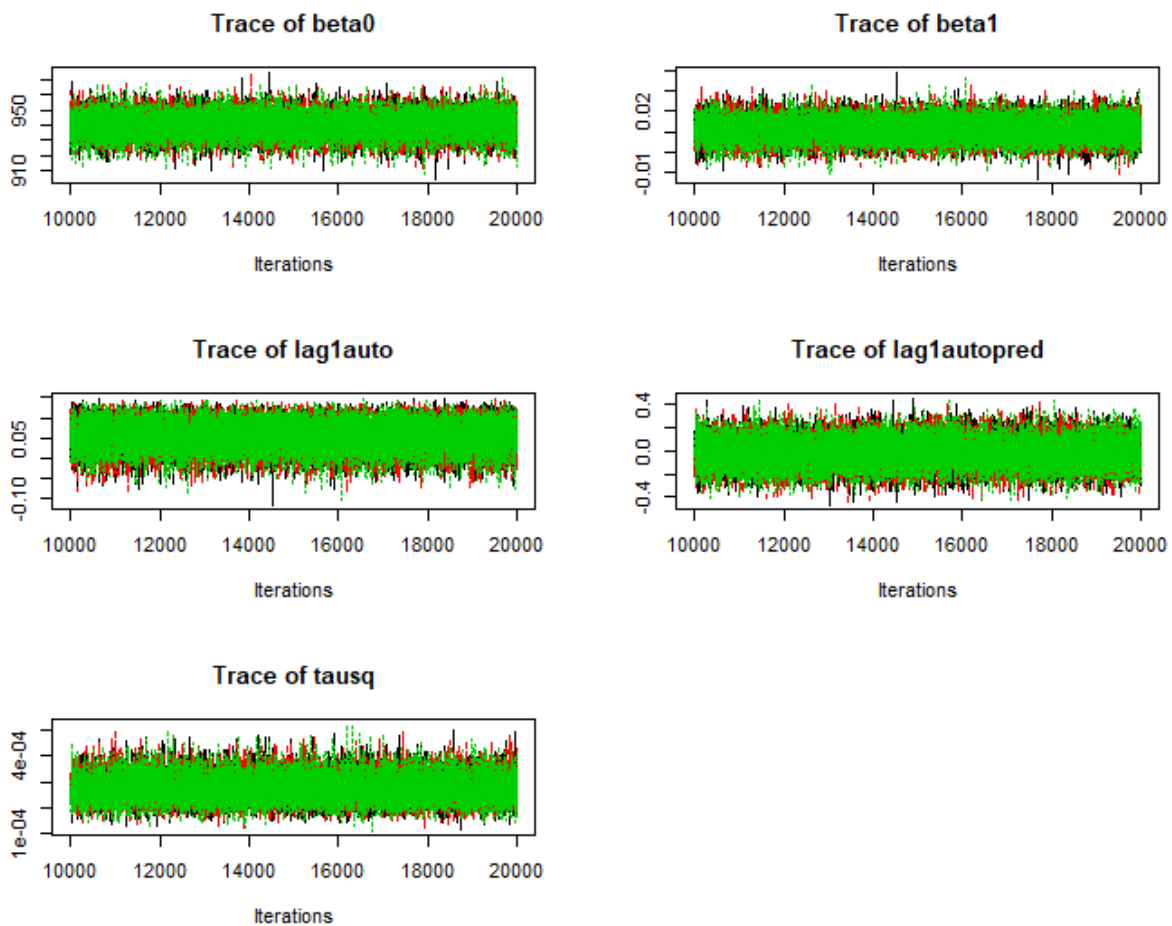
Analysis using Model with one predictor variable:

See appendix 3) for Model definition in R

1) Using population density variable:

See appendix 4) for using population density variable in R, 5) for Glm summary
6) for initial values for the parameters in R

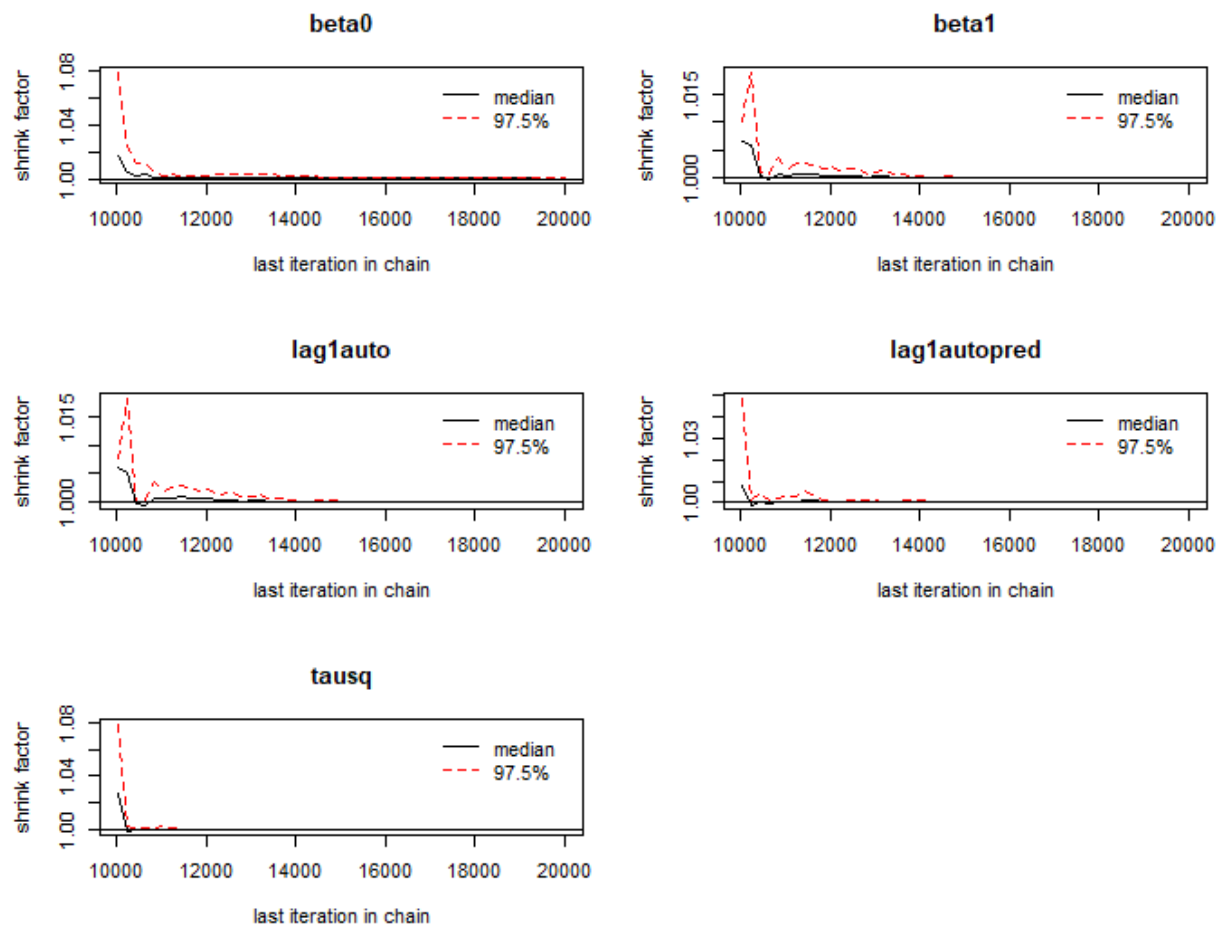
1.1) Traceplot:



Traceplot for all the parameters looks great, plots for all the parameters looks like white noise.

Note: beta0 is the intercept, beta1 is the slope, lag1auto is the lag1 autocorrelation of the residuals of the real dataset, lag1autopred is the lag1 autocorrelation of the residuals of the replicated datasets and tausq is the precision.

1.2) Gelman plot:



Shrink factor for every parameter goes down to 1 immediately after specified number of iterations (10000). Shrink factor values for each parameter start from way below of 1.2.

1.3) Posterior Summary:

```

Iterations = 20001:30000
Thinning interval = 1
Number of chains = 3
Sample size per chain = 10000

```

1. Empirical mean and standard deviation for each variable,
plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
beta0	9.403e+02	7.944e+00	4.586e-02	4.653e-02
beta1	1.137e-02	5.500e-03	3.176e-05	3.191e-05
lag1auto	5.585e-02	3.633e-02	2.097e-04	2.097e-04
lag1autopred	-1.697e-02	1.267e-01	7.317e-04	7.209e-04
pppval	7.070e-01	4.552e-01	2.628e-03	2.624e-03
tausq	2.735e-04	5.059e-05	2.921e-07	2.984e-07

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta0	9.245e+02	9.350e+02	9.403e+02	9.456e+02	9.557e+02
beta1	5.209e-04	7.742e-03	1.136e-02	1.507e-02	2.212e-02
lag1auto	-2.015e-02	3.167e-02	5.819e-02	8.205e-02	1.197e-01
lag1autopred	-2.638e-01	-1.041e-01	-1.749e-02	7.008e-02	2.305e-01
pppval	0.000e+00	0.000e+00	1.000e+00	1.000e+00	1.000e+00
tausq	1.834e-04	2.377e-04	2.705e-04	3.058e-04	3.807e-04

1.4) DIC values:

```

> logit.dic.pd
Mean deviance: 663.6
penalty 3.111
Penalized deviance: 666.7
> logit.dic.popt
Mean deviance: 663.6
penalty 6.541
Penalized deviance: 670.2

```

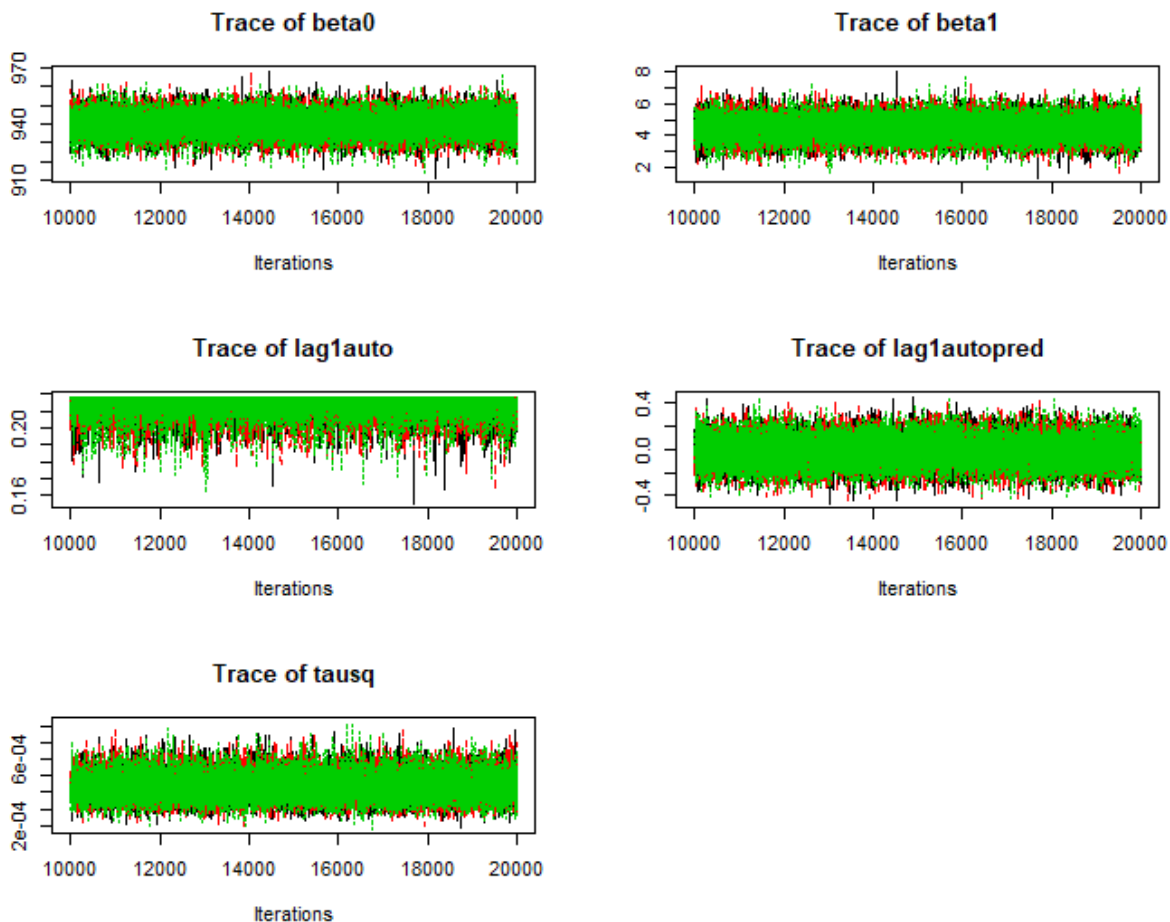
1.5) Analysis:

Larger values of population density will cause small increase in the death rate since, the value of slope is positive and the credible sets for the slope does not have zero. Not surprisingly, lag1autopred (lag 1 autocorrelation in the residuals from the replicated datasets) has a mean very close to 0 and 0 is near the center of 95% credible set. The lag1auto (lag 1 autocorrelation in the residuals from the real dataset) has a mean close to 0 but farther from 0 as compared to mean of lag1autopred and its credible set also contains 0 between first two quarters. Most importantly, predictive p-values is 0.7 approx. indicating that lag1auto was larger than lag1autopred in about 70% of the replications. Although value close to 0.5 is preferred but 0.7 is not extreme, the real data is not a typical of data generated from this model. A value greater than 0.9 would have suggested that we need to expand the model to account for autocorrelation in the data.

2) Using percentage of non-white population variable:

See appendix 7) for using percentage of non-white population variable in R, 8) for Glm summary 9) for initial values for the parameters in R

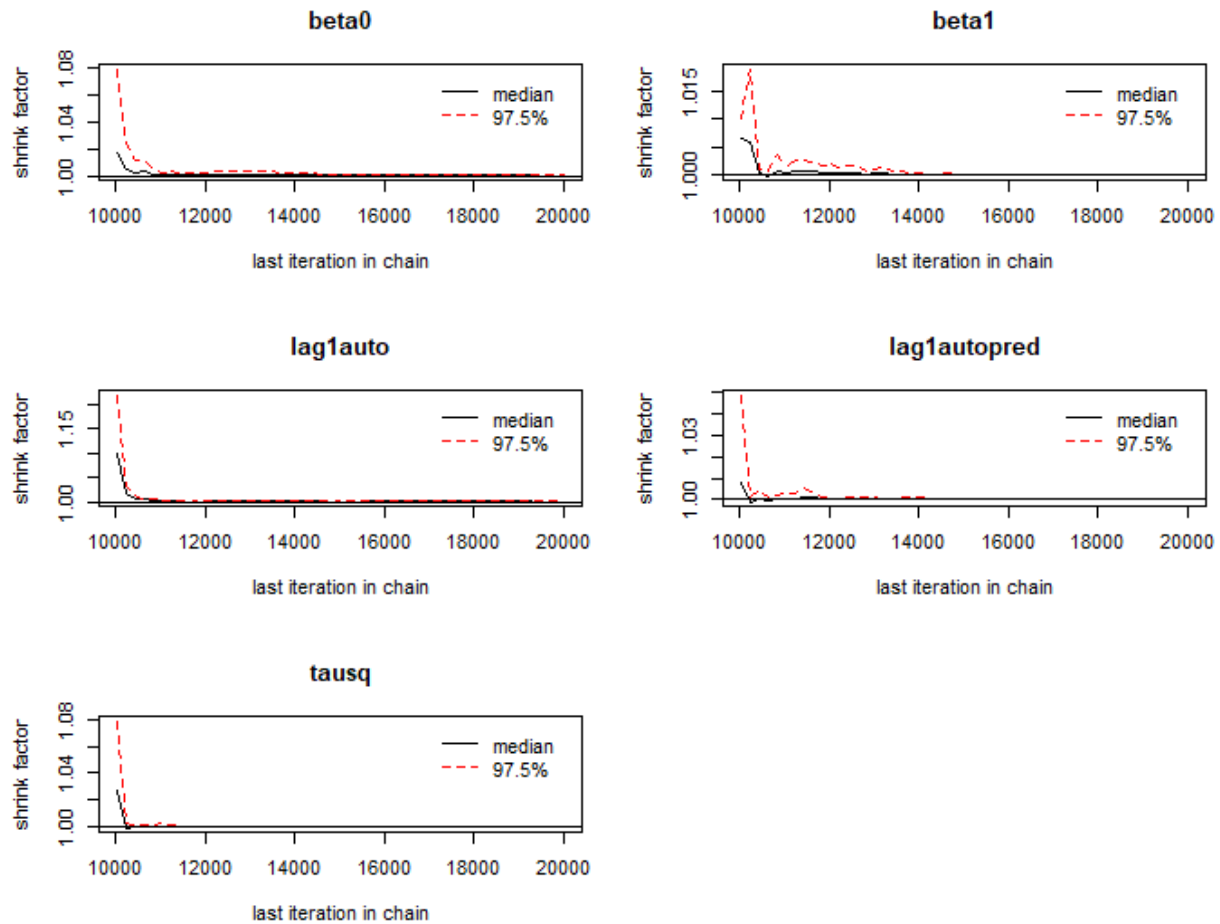
2.1) Traceplot:



Traceplot for all the parameters looks great, plots for all the parameters looks like white noise.

Note: beta0 is the intercept, beta1 is the slope, lag1auto is the lag1 autocorrelation of the residuals of the real dataset, lag1autopred is the lag1 autocorrelation of the residuals of the replicated datasets and tausq is the precision.

2.3) Gelman plot:



Shrink factor for every parameter goes down to 1 immediately after specified number of iterations (10000). Shrink factor values for most parameters start from way below of 1.2 and values for lag1auto starts from close to 1.2.

2.4) Posterior Summary:

```

Iterations = 20001:30000
Thinning interval = 1
Number of chains = 3
Sample size per chain = 10000

```

1. Empirical mean and standard deviation for each variable,
plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
beta0	9.403e+02	6.305e+00	3.640e-02	3.693e-02
beta1	4.491e+00	7.117e-01	4.109e-03	4.129e-03
lag1auto	2.143e-01	5.628e-03	3.249e-05	3.282e-05
lag1autopred	-1.697e-02	1.267e-01	7.317e-04	7.209e-04
pppval	9.658e-01	1.818e-01	1.050e-03	1.057e-03
tausq	4.341e-04	8.031e-05	4.637e-07	4.736e-07

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta0	9.278e+02	9.361e+02	9.403e+02	9.445e+02	9.525e+02
beta1	3.087e+00	4.022e+00	4.490e+00	4.969e+00	5.882e+00
lag1auto	1.984e-01	2.130e-01	2.165e-01	2.180e-01	2.185e-01
lag1autopred	-2.638e-01	-1.041e-01	-1.749e-02	7.008e-02	2.305e-01
pppval	0.000e+00	1.000e+00	1.000e+00	1.000e+00	1.000e+00
tausq	2.911e-04	3.773e-04	4.294e-04	4.854e-04	6.043e-04

2.5) DIC values:

```

> logit.dic.pd
Mean deviance: 635.9
penalty 3.111
Penalized deviance: 639
> logit.dic.popt
Mean deviance: 635.9
penalty 6.519
Penalized deviance: 642.4

```

2.6) Analysis:

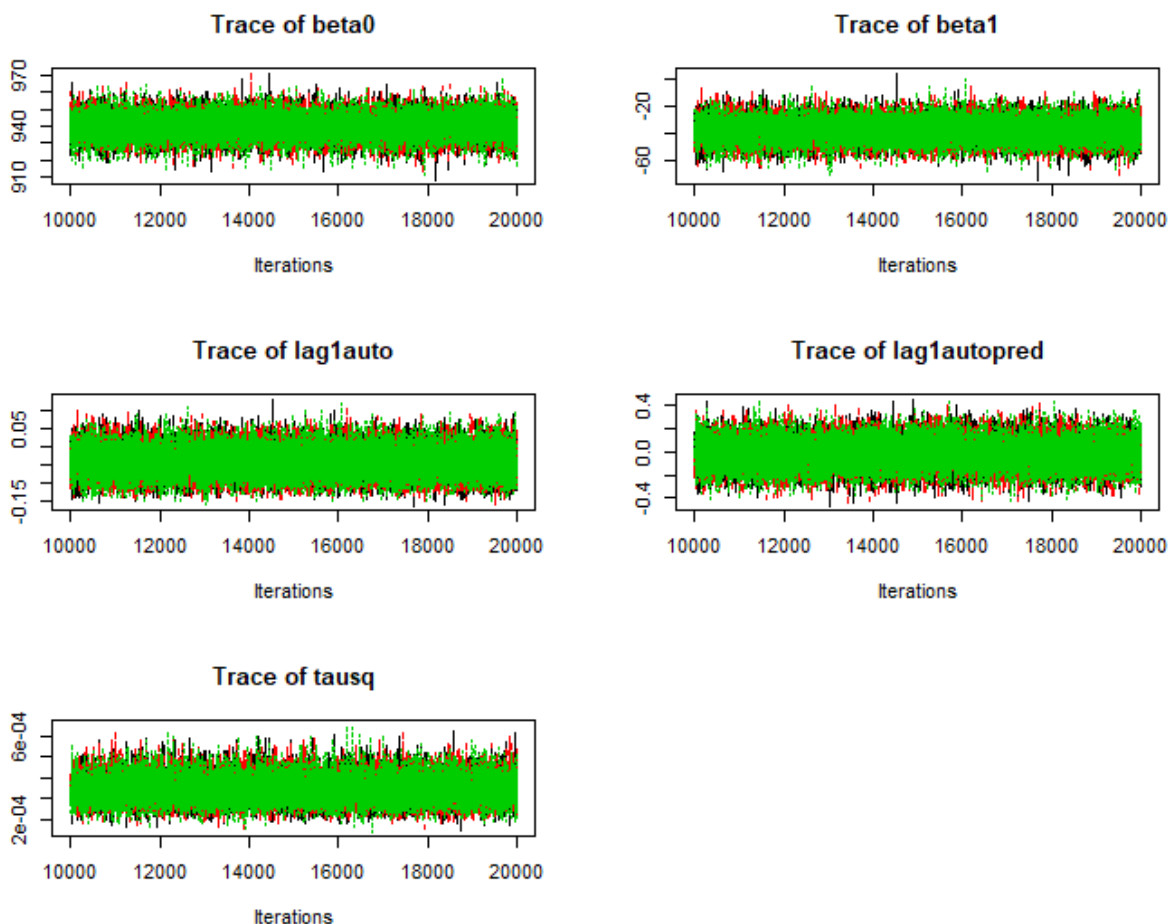
Larger values of percentage of non-white population will cause greater increase in the death rate since, the value of slope is positive and large. The credible sets for the slope does not have zero. lag1autopred (lag 1 autocorrelation in the residuals from the replicated datasets) has a mean very close to 0 and 0 is near the center of 95% credible set. The lag1auto (lag 1 autocorrelation in the residuals from the real dataset) has a mean close to 0 but farther from 0 as compared to mean of lag1autopred and its credible set does not contain 0. Most importantly, predictive p-values is 0.9 approx. indicating that lag1auto was larger than lag1autopred in about 90% of the replications. Its value close to 0.5 is preferred but 0.9 is extreme, which suggests that we need to expand the model to account for

autocorrelation in the data. However, the DIC values obtained for this model is way less than the values obtained from model with population density variable.

3) Using average number of years of education of adults variable

See appendix 10) for using numbers of years of education of adults in R, 11) for Glm summary 12) for initial values for the parameters in R

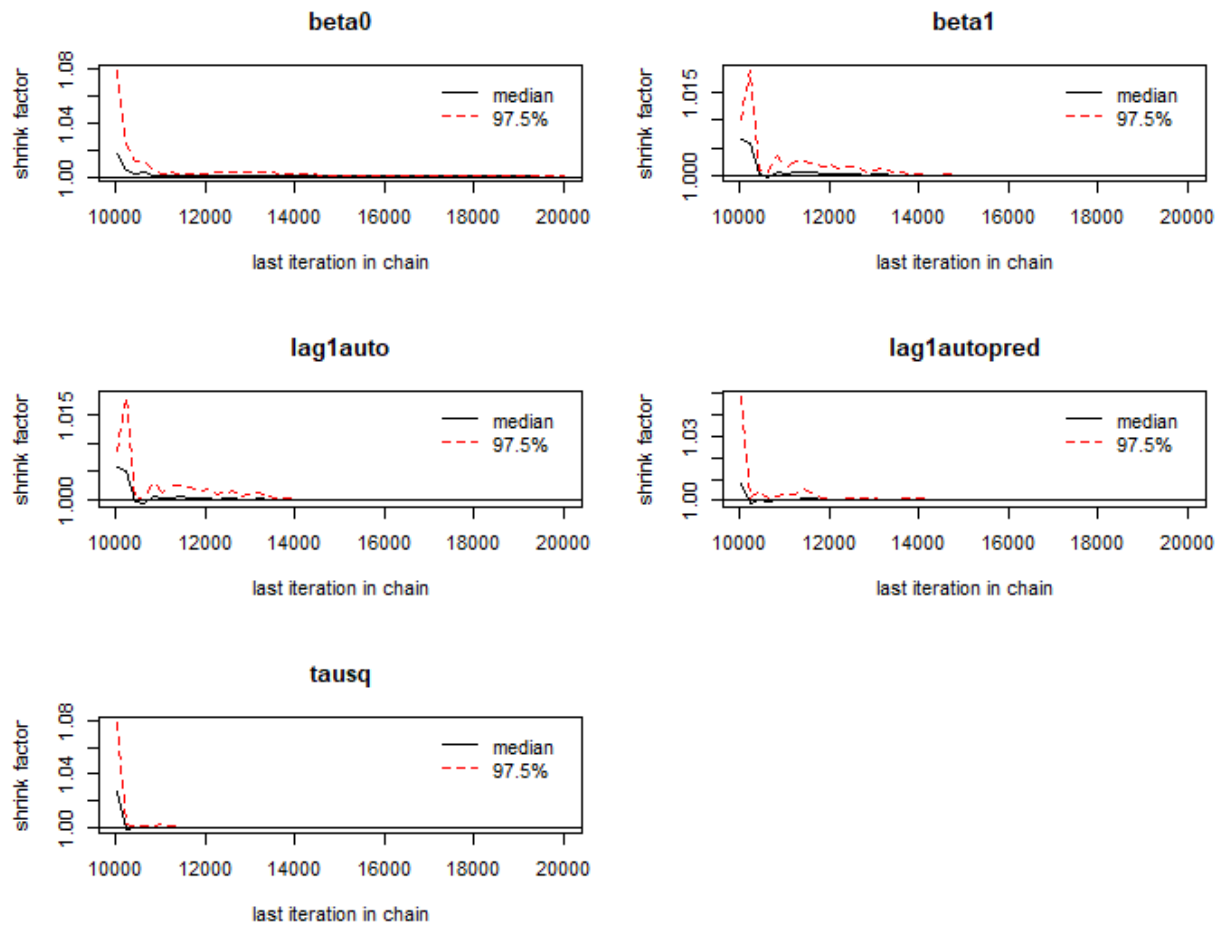
3.2) Traceplot:



Traceplot for all the parameters looks great, plots for all the parameters look like white noise.

Note: beta0 is the intercept, beta1 is the slope, lag1auto is the lag1 autocorrelation of the residuals of the real dataset, lag1autopred is the lag1 autocorrelation of the residuals of the replicated datasets and tausq is the precision.

3.3) Gelman plot:



Shrink factor for every parameter converges to 1 immediately after specified number of iterations (10000) and the shrink factor values for each parameter start from way below of 1.2.

3.4) Posterior Summary:

```

Iterations = 20001:30000
Thinning interval = 1
Number of chains = 3
Sample size per chain = 10000

```

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
beta0	940.275580	7.082e+00	4.089e-02	4.149e-02
beta1	-37.579007	8.435e+00	4.870e-02	4.894e-02
lag1auto	-0.041242	4.146e-02	2.393e-04	2.393e-04
lag1autopred	-0.016967	1.267e-01	7.317e-04	7.209e-04
pppval	0.432167	4.954e-01	2.860e-03	2.795e-03
tausq	0.000344	6.365e-05	3.675e-07	3.753e-07

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta0	9.262e+02	935.549727	9.403e+02	9.450e+02	9.540e+02
beta1	-5.422e+01	-43.148106	-3.759e+01	-3.191e+01	-2.109e+01
lag1auto	-1.179e-01	-0.070537	-4.267e-02	-1.292e-02	4.227e-02
lag1autopred	-2.638e-01	-0.104141	-1.749e-02	7.008e-02	2.305e-01
pppval	0.000e+00	0.000000	0.000e+00	1.000e+00	1.000e+00
tausq	2.307e-04	0.000299	3.403e-04	3.847e-04	4.789e-04

3.5) DIC values:

```

> logit.dic.pd
Mean deviance: 649.9
penalty 3.111
Penalized deviance: 653
> logit.dic.popt
Mean deviance: 649.8
penalty 6.481
Penalized deviance: 656.3

```

3.6) Analysis:

Larger values of average number of years of education of adults will cause greater decrease in the death rate since, the value of slope is negative and way below 0. The credible sets for the slope does not have zero. lag1autopred (lag 1 autocorrelation in the residuals from the replicated datasets) has a mean very close to 0 and 0 is near the center of 95% credible set. The lag1auto (lag 1 autocorrelation in the residuals from the real dataset) has a mean close to 0 but farther from 0 as compared to mean of lag1autopred and its credible set also contains 0 between last two quarters. Most importantly, predictive p-values is 0.43 approx. indicating that lag1auto was larger than lag1autopred in about 43% of the replications. Value close to 0.5 is preferred and 0.43 is closer to it, the real data falls near the center of lag 1 autocorrelation in datasets drawn from the posterior predictive distribution of the

model we fit. There is no evidence here that the assumption of independent errors is violated. The DIC values for this model is smaller than the model with population density variable but greater than the model with percentage of non-white population.

Overall Analysis after evaluation of one parameter models:

With the help of one parameter models, we found out that population density and percentage of non white population variables correlates positively with death rate. However, increase in percentage of non-white population causes significant increase in death rate this gives us the notion that in metropolitan area of US, the non-white population are prone to some terminating illnesses or they have weaker immune system as compared to white population. Number of years of education of adults correlates negatively with death rate. This makes sense as with education people tend to learn more about healthy lifestyle and they get acquainted with proper health measures which they take throughout their life. In terms of DIC value the model with percentage of non white population variable appears to be doing better than other however, in terms of predictive checking this model does not perform good, in this case, model with number of years of education of adults performs way better. So, it turns out percentage of non white population and number of years of education of adults are important predictor variables for estimating death rate.

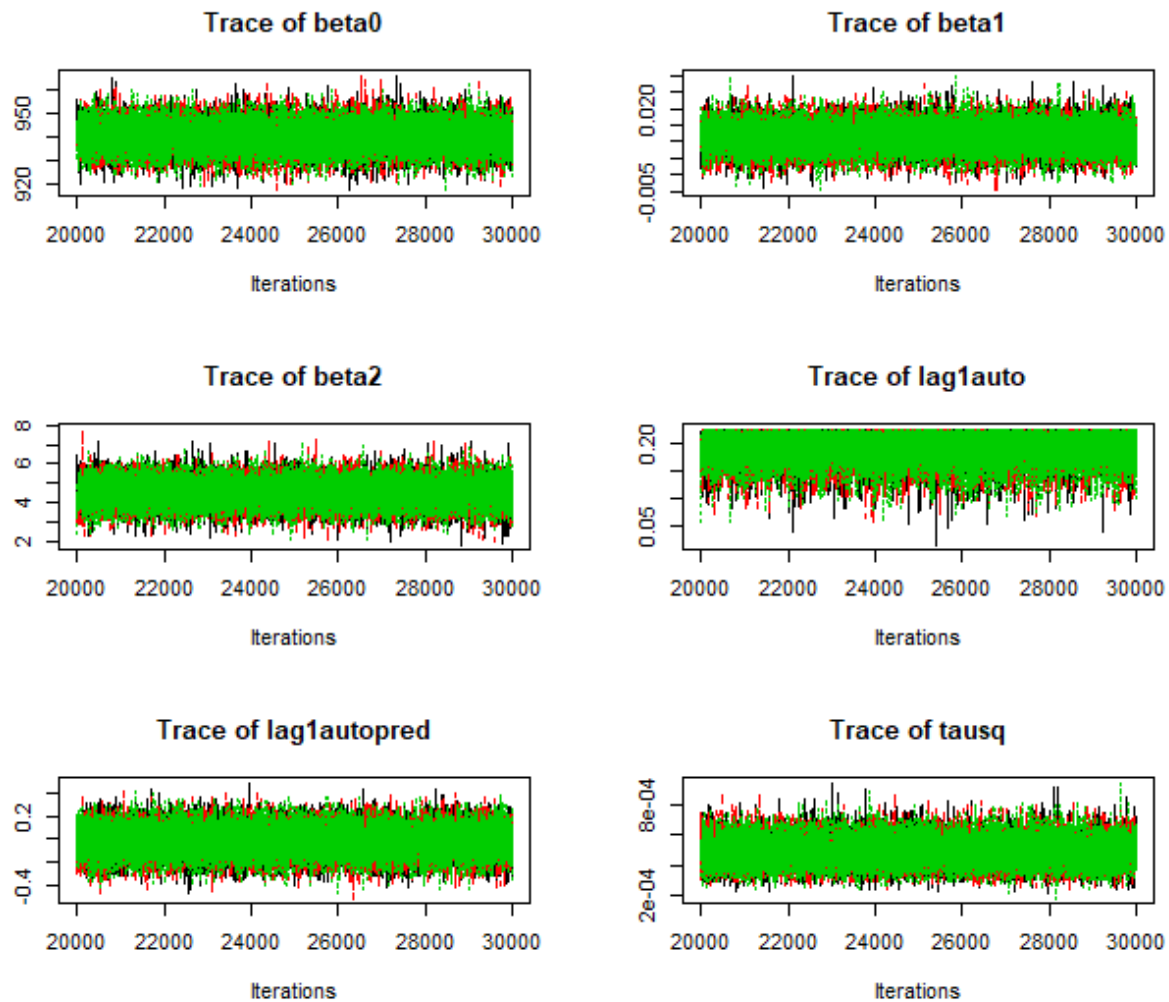
Analysis using Model with two predictors variable:

See appendix 13) for Model definition in R.

1) Using population density and percentage of non-white population variables:

See appendix 14) for using these variables in R, 15) for Glm summary 16) for initial values for the parameters in R.

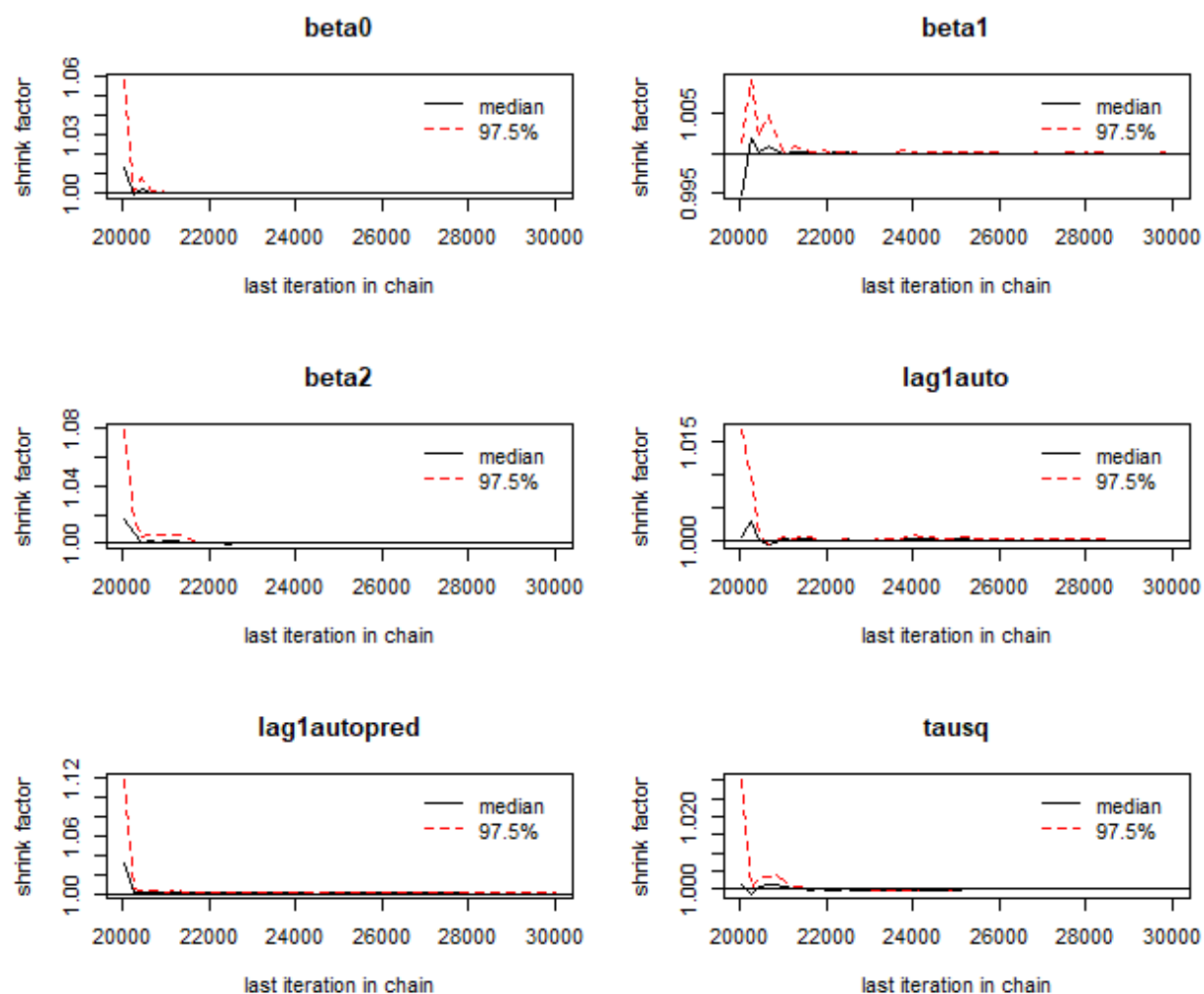
1.1) Traceplot:



Traceplot for all the parameters looks great, plots for all the parameters looks like white noise.

Note: beta0 is the intercept, beta1 is the slope of population density variable, beta2 is the slope of percentage of non white population variable, lag1auto is the lag1 autocorrelation of the residuals of the real dataset, lag1autopred is the lag1 autocorrelation of the residuals of the replicated datasets and tausq is the precision.

1.2) Gelman plot:



Shrink factor for every parameter converges to 1 immediately after specified number of iterations (10000). Shrink factor values for most parameter start from way below of 1.2 and values for lag1autopred starts from close to 1.2.

1.3) Posterior Summary:

```

Iterations = 30001:40000
Thinning interval = 1
Number of chains = 3
Sample size per chain = 10000

```

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
beta0	9.403e+02	5.958e+00	3.440e-02	3.478e-02
beta1	1.151e-02	4.152e-03	2.397e-05	2.397e-05
beta2	4.505e+00	6.704e-01	3.871e-03	3.849e-03
lag1auto	1.944e-01	2.608e-02	1.506e-04	1.522e-04
lag1autopred	-1.688e-02	1.260e-01	7.277e-04	7.327e-04
pppval	9.490e-01	2.200e-01	1.270e-03	1.276e-03
tausq	4.866e-04	9.121e-05	5.266e-07	5.471e-07

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta0	9.285e+02	9.364e+02	9.403e+02	9.443e+02	9.520e+02
beta1	3.412e-03	8.714e-03	1.150e-02	1.427e-02	1.974e-02
beta2	3.185e+00	4.058e+00	4.506e+00	4.951e+00	5.827e+00
lag1auto	1.302e-01	1.804e-01	2.005e-01	2.146e-01	2.255e-01
lag1autopred	-2.627e-01	-1.026e-01	-1.781e-02	6.911e-02	2.312e-01
pppval	0.000e+00	1.000e+00	1.000e+00	1.000e+00	1.000e+00
tausq	3.252e-04	4.223e-04	4.804e-04	5.439e-04	6.812e-04

1.4) DIC values:

```

> logit.dic.pd
Mean deviance: 629
penalty 4.2
Penalized deviance: 633.2
> logit.dic.popt
Mean deviance: 629
penalty 8.732
Penalized deviance: 637.7

```

1.5) Analysis:

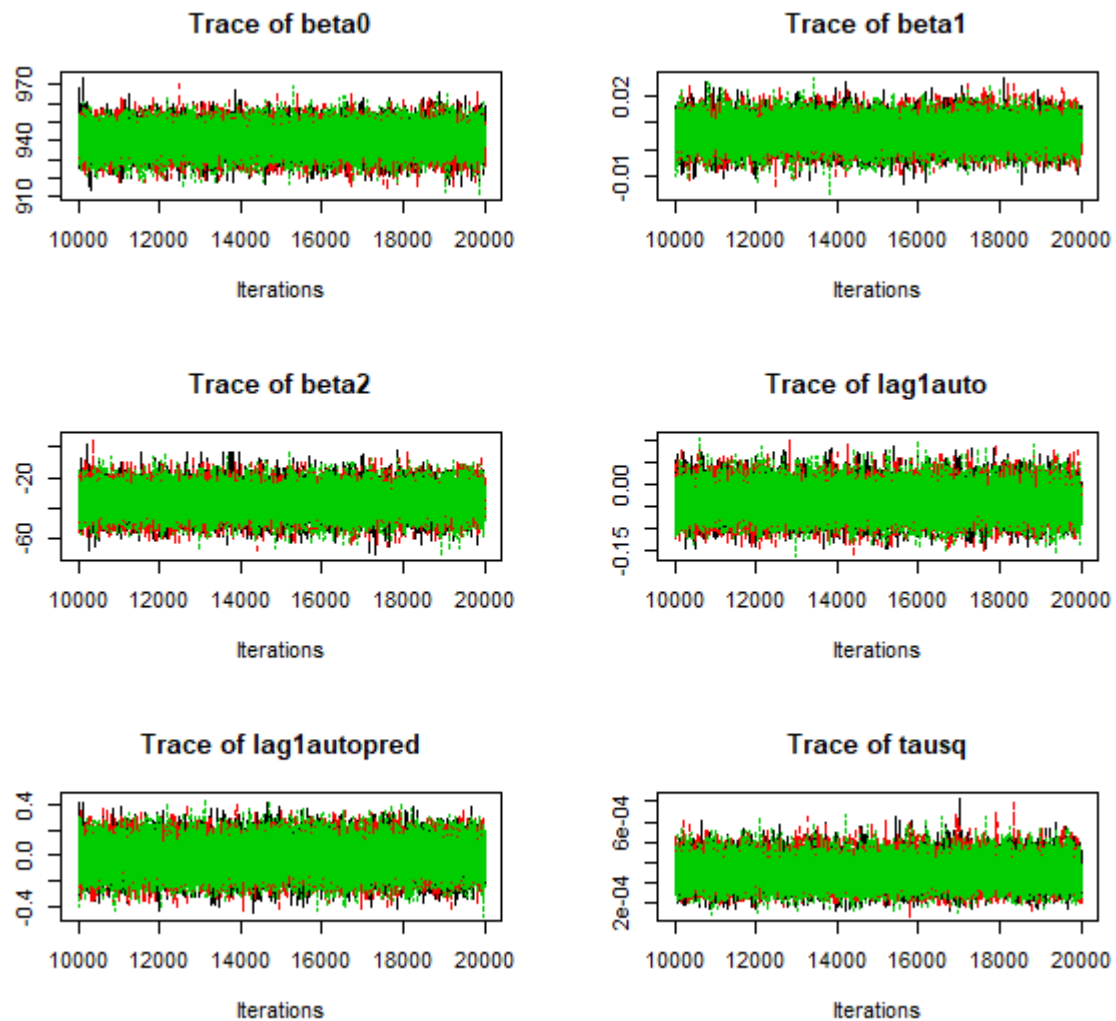
Larger values of population density will cause small increase in the death rate since, the value of slope is positive and the credible sets for its slope does not have zero. Larger values of percentage of non white population will cause great increase in the death rate since, the value of slope is positive and far away from 0. The credible sets for its slope does not have zero. lag1autopred (lag 1 autocorrelation in the residuals from the replicated datasets) has a mean very close to 0 and 0 is near the center of 95% credible set. The lag1auto (lag 1 autocorrelation in the residuals from the real dataset) has a mean close to 0 but farther from 0 as compared to mean of lag1autopred and its credible set does not

contain 0. Most importantly, predictive p-values is 0.9 approx. indicating that lag1auto was larger than lag1autopred in about 90% of the replications. Although value close to 0.5 is preferred but 0.9 is extreme, this suggests that we need to expand the model to account for autocorrelation in the data. Also, DIC values for this model is lowest than all the one parameter models.

2) Using population density and number of years of education of adults variables:

See appendix 17) for using these variables in R, 18) for Glm summary 19) for initial values for the parameters in R.

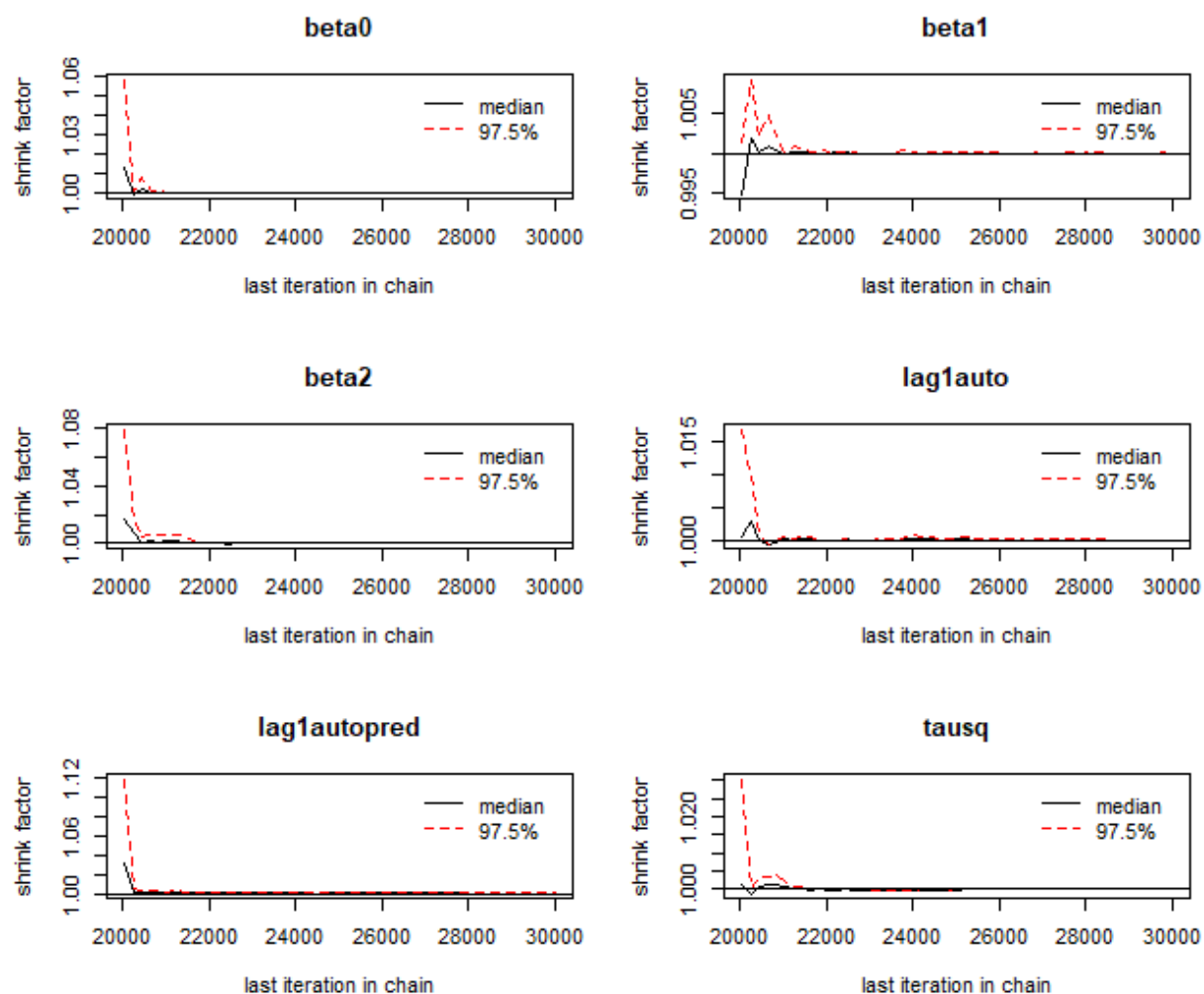
2.1) Traceplot:



Traceplot for all the parameters looks great, plots for all the parameters looks like white noise.

Note: beta0 is the intercept, beta1 is the slope of population density variable, beta2 is the slope of percentage of non white population variable, lag1auto is the lag1 autocorrelation of the residuals of the real dataset, lag1autopred is the lag1 autocorrelation of the residuals of the replicated datasets and tausq is the precision.

2.2) Gelman plot:



Shrink factor for every parameter converges to 1 immediately after double the amount of specified number of iterations (10000). Shrink factor values for most parameter start from way below of 1.2 and values for lag1autopred starts from close to 1.2.

2.3) Posterior Summary:

```

Iterations = 40001:50000
Thinning interval = 1
Number of chains = 3
Sample size per chain = 10000

```

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
beta0	940.340157	7.065e+00	4.079e-02	4.079e-02
beta1	0.006412	5.040e-03	2.910e-05	2.910e-05
beta2	-34.974513	8.576e+00	4.951e-02	4.898e-02
lag1auto	-0.043194	3.393e-02	1.959e-04	1.947e-04
lag1autopred	-0.017088	1.267e-01	7.313e-04	7.403e-04
pppval	0.421400	4.938e-01	2.851e-03	2.851e-03
tausq	0.000348	6.489e-05	3.746e-07	3.954e-07

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta0	9.265e+02	935.653318	9.404e+02	9.450e+02	9.543e+02
beta1	-3.466e-03	0.003039	6.382e-03	9.767e-03	1.632e-02
beta2	-5.185e+01	-40.701381	-3.496e+01	-2.928e+01	-1.820e+01
lag1auto	-1.050e-01	-0.066748	-4.521e-02	-2.155e-02	2.893e-02
lag1autopred	-2.654e-01	-0.103894	-1.680e-02	7.021e-02	2.298e-01
pppval	0.000e+00	0.000000	0.000e+00	1.000e+00	1.000e+00
tausq	2.326e-04	0.000302	3.436e-04	3.893e-04	4.859e-04

2.4) DIC values:

```

> logit.dic.pd
Mean deviance: 649.2
penalty 4.198
Penalized deviance: 653.4
> logit.dic.popt
Mean deviance: 649.1
penalty 8.546
Penalized deviance: 657.7

```

2.5) Analysis:

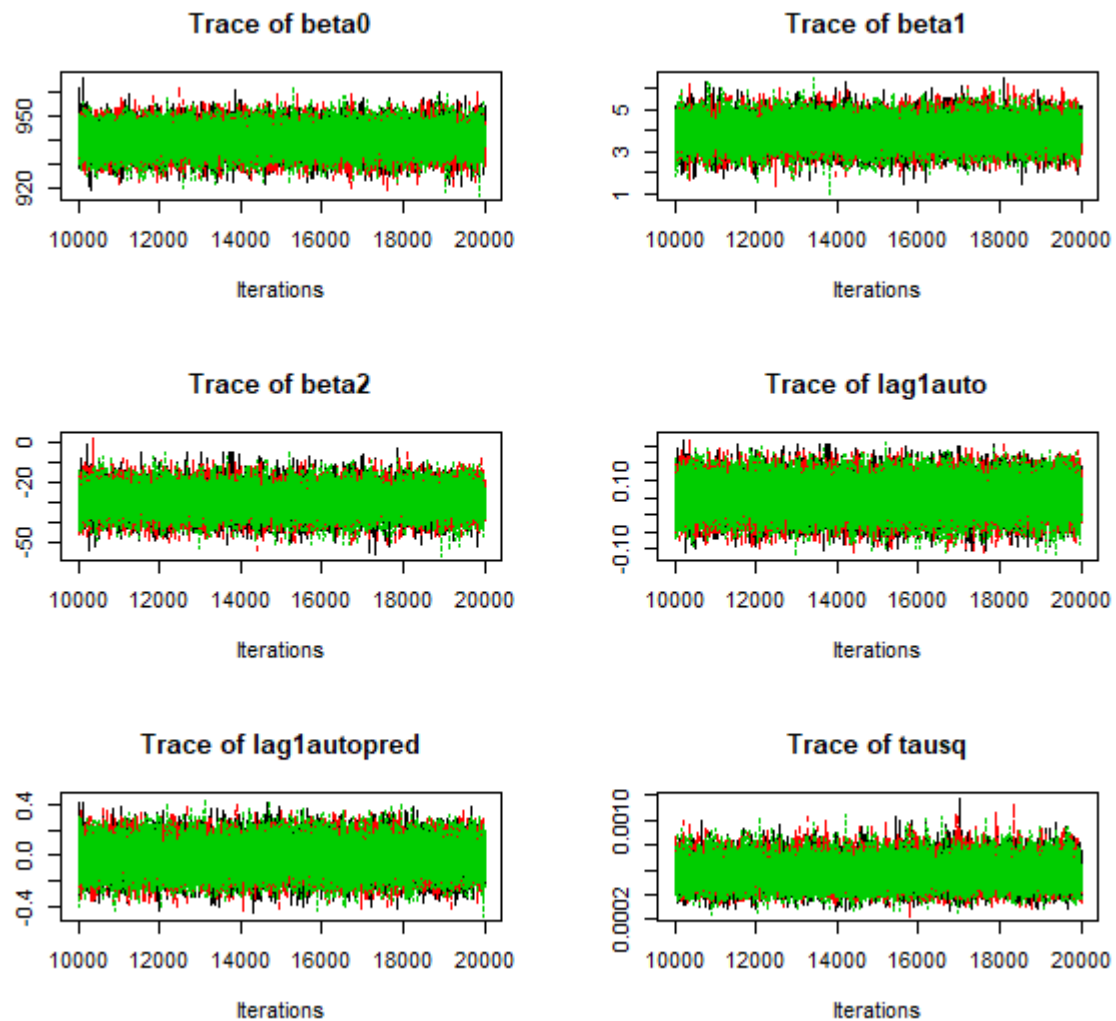
Larger values of population density will cause small increase in the death rate since, the value of slope is positive and the credible sets for its slope does not have zero. Larger values of number of years of education of adults will cause great decrease in the death rate since, the value of slope is negative and far away from 0. The credible sets for its slope does not have zero. lag1autopred (lag 1 autocorrelation in the residuals from the replicated datasets) has a mean very close to 0 and 0 is near the center of 95% credible set. The lag1auto (lag 1 autocorrelation in the residuals from the real dataset) has a mean close to 0 but farther from 0 as compared to mean of lag1autopred and its credible set contains 0

between last two quarters. Most importantly, predictive p-values is 0.42 approx. indicating that lag1auto was larger than lag1autopred in about 42% of the replications. The value close to 0.5 is preferred and 0.42 is close to it, this suggests that the real data falls near the center of lag 1 autocorrelation in datasets drawn from the posterior predictive distribution of the model we fit and there is no evidence here that the assumption of independent errors is violated. The DIC values for this model are larger than the model with population density and percentage of non white population variables.

3) Using percentage of non white population and number of years of education of adults variables:

See appendix 20) for using these variables in R, 21) for Glm summary 22) for initial values for the parameters in R.

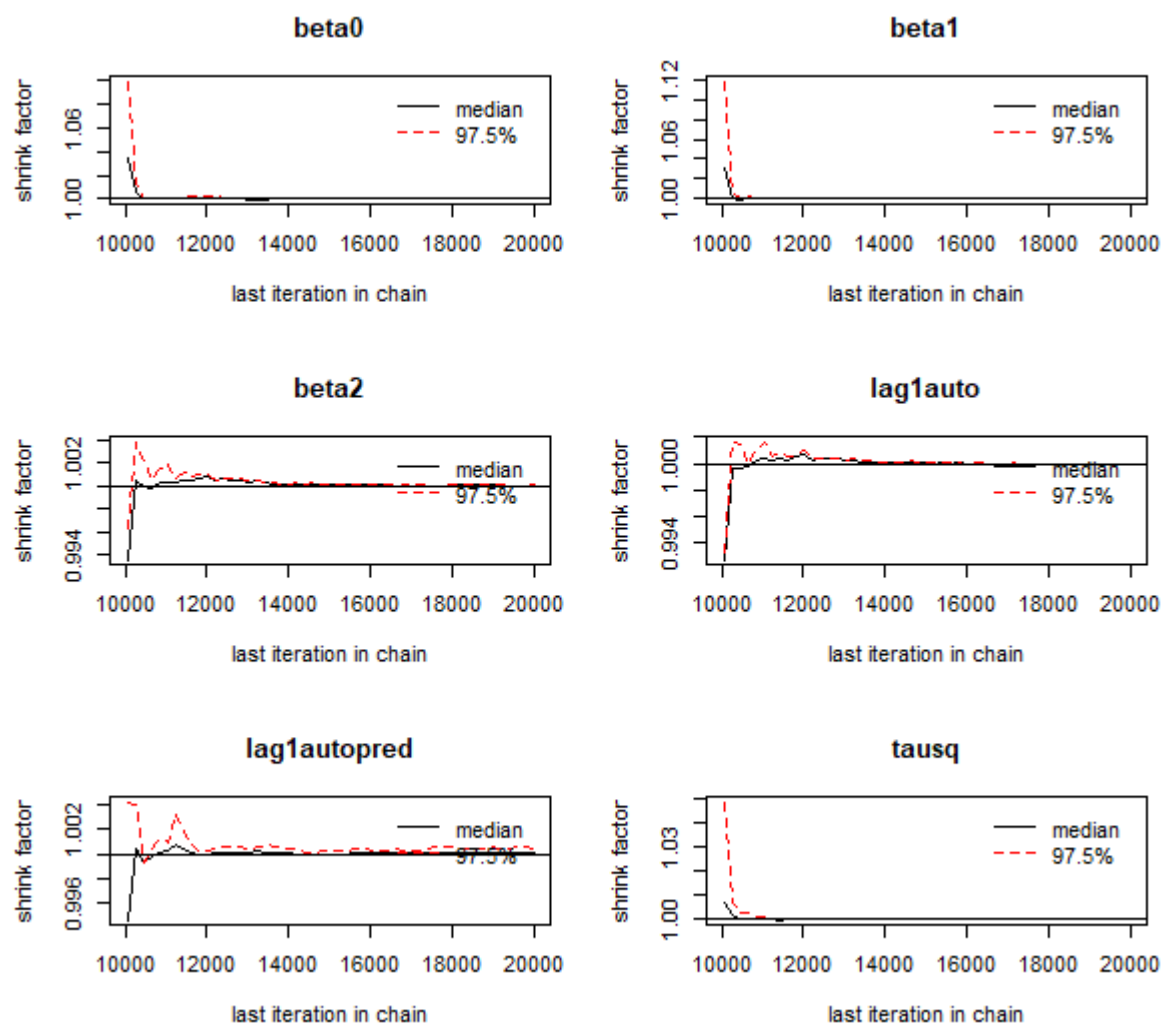
3.1) Traceplot:



Traceplot for all the parameters looks great, plots for all the parameters looks like white noise.

Note: beta0 is the intercept, beta1 is the slope of population density variable, beta2 is the slope of percentage of non white population variable, lag1auto is the lag1 autocorrelation of the residuals of the real dataset, lag1autopred is the lag1 autocorrelation of the residuals of the replicated datasets and tausq is the precision.

3.2) Gelman plot:



Shrink factor for every parameter converges to 1 immediately after specified number of iterations (10000). Shrink factor values for most parameter start from way below of 1.2 and values for beta1 starts from close to 1.2.

3.3) Posterior Summary:

```

Iterations = 26001:36000
Thinning interval = 1
Number of chains = 3
Sample size per chain = 10000

```

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
beta0	9.404e+02	5.5296052	3.193e-02	3.211e-02
beta1	3.914e+00	0.6365192	3.675e-03	3.807e-03
beta2	-2.899e+01	6.7333795	3.888e-02	3.868e-02
lag1auto	5.480e-02	0.0503999	2.910e-04	2.910e-04
lag1autopred	-1.681e-02	0.1277443	7.375e-04	7.375e-04
pppval	6.987e-01	0.4588304	2.649e-03	2.649e-03
tausq	5.708e-04	0.0001061	6.126e-07	6.454e-07

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta0	9.296e+02	9.367e+02	9.403e+02	9.440e+02	9.513e+02
beta1	2.661e+00	3.494e+00	3.913e+00	4.339e+00	5.164e+00
beta2	-4.221e+01	-3.351e+01	-2.894e+01	-2.447e+01	-1.596e+01
lag1auto	-4.493e-02	1.978e-02	5.638e-02	9.022e-02	1.482e-01
lag1autopred	-2.663e-01	-1.037e-01	-1.690e-02	6.969e-02	2.336e-01
pppval	0.000e+00	0.000e+00	1.000e+00	1.000e+00	1.000e+00
tausq	3.811e-04	4.964e-04	5.639e-04	6.383e-04	7.954e-04

3.4) DIC values:

```

> logit.dic.pd
Mean deviance: 619.4
penalty 4.072
Penalized deviance: 623.5
> logit.dic.popt
Mean deviance: 619.4
penalty 8.867
Penalized deviance: 628.3

```

3.5) Analysis:

Larger values of percentage of non white population will cause great increase in the death rate since, the value of slope is positive and far away from 0. The credible sets for its slope does not have zero. Larger values of number of years of education of adults will cause great decrease in the death rate since, the value of slope is negative and far away from 0. The credible sets for its slope does not have zero. lag1autopred (lag 1 autocorrelation in the residuals from the replicated datasets) has a mean very close to 0 and 0 is near the center of 95% credible set. The lag1auto (lag 1 autocorrelation in the residuals from the real dataset) has a mean close to 0 but farther from 0 as compared to mean of lag1autopred and its credible set contains 0 between first two quarters. Most importantly, predictive p-values

is 0.7 approx. indicating that lag1auto was larger than lag1autopred in about 70% of the replications. Although value close to 0.5 is preferred but 0.7 is not extreme, the real data is not a typical of data generated from this model. A value greater than 0.9 would have suggested that we need to expand the model to account for autocorrelation in the data. The DIC values for this model is smaller than all the models we have discussed so far.

Overall Analysis after evaluation of two and one parameters models:

With the help of two parameters model, we actually confirm the results of analysis of one parameter models that percentage of non white population and number of years of education of adults are most important predictor variables for estimating death rate. That is why when we combine these two variables in two parameter models we get better results especially in terms of DIC. However, in terms of predictive checking two parameter mode with number of years of education of adults and population density variables performs way better. This gives us the idea that population density alone may not be the good predictor but it becomes important when combined with other variable with similar relation with response variable.

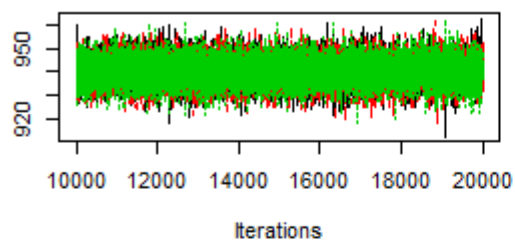
Analysis using Model with three predictors variable:

See appendix 23) for model definition in R.

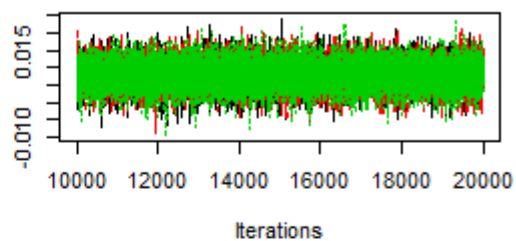
See appendix 24) for using these variables in R, 25) for Glm summary 26) for initial values for the parameters in R.

Traceplot:

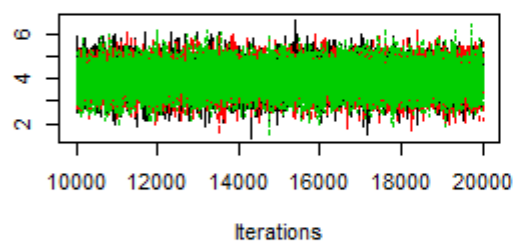
Trace of beta0



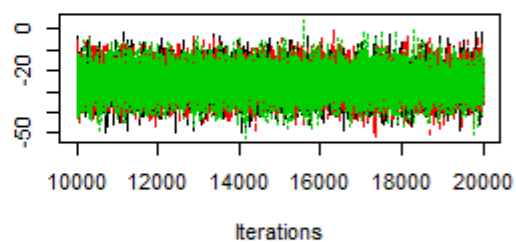
Trace of beta1



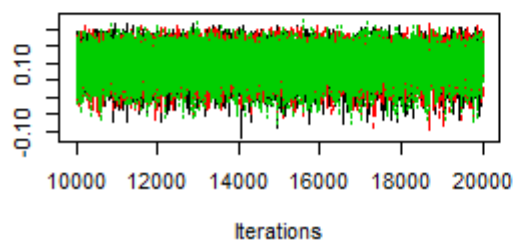
Trace of beta2



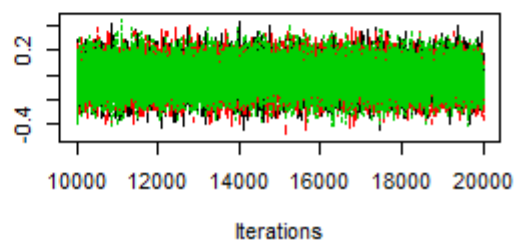
Trace of beta3

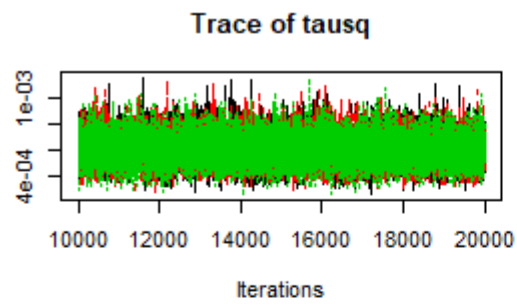


Trace of lag1auto



Trace of lag1autopred

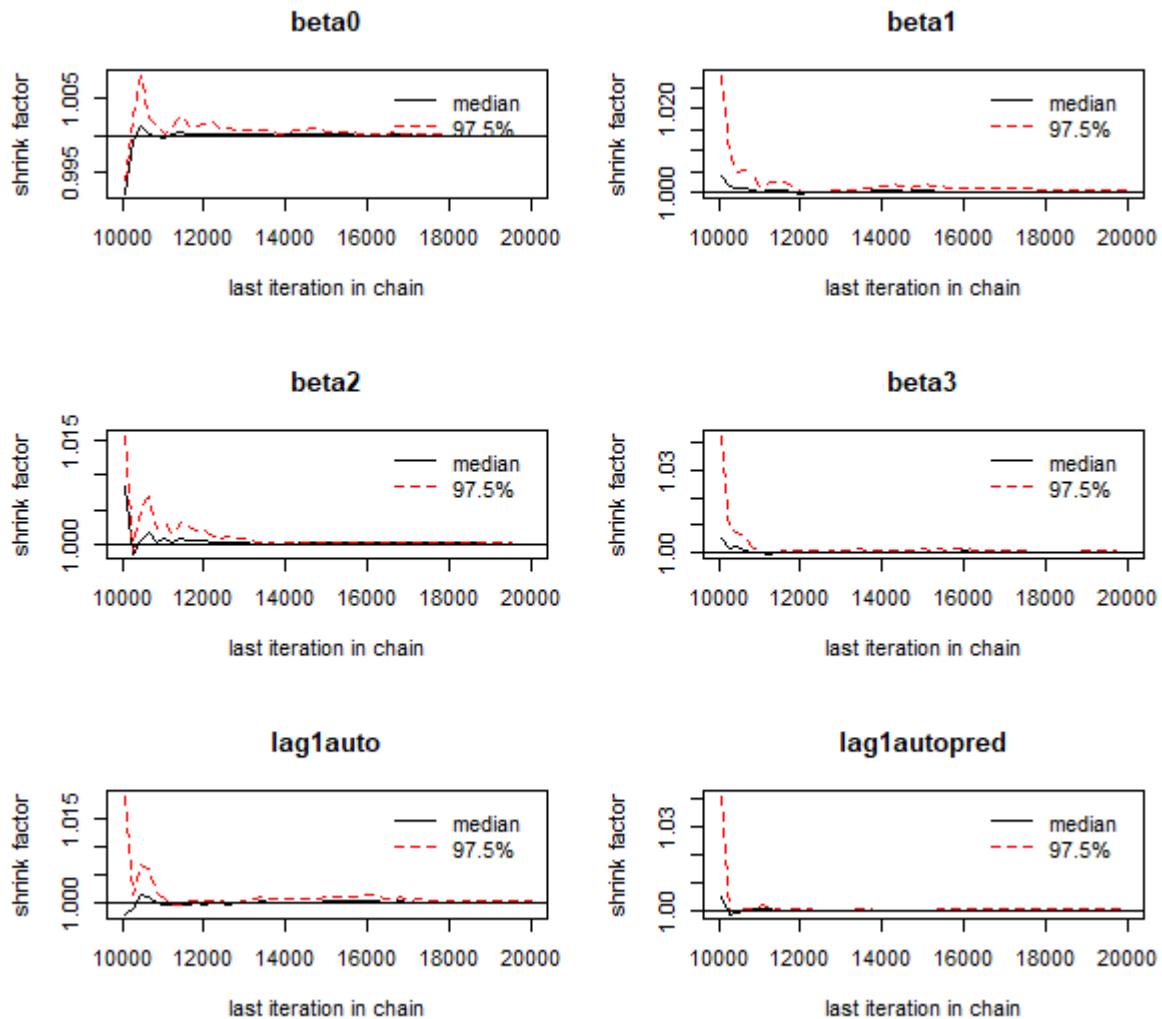




Traceplot for all the parameters looks great, plots for all the parameters looks like white noise.

Note: β_0 is the intercept, β_1 is the slope of population density variable, β_2 is the slope of percentage of non white population variable, β_3 is the slope of percentage of non white population, $\text{lag1}_{\text{auto}}$ is the lag1 autocorrelation of the residuals of the real dataset, $\text{lag1}_{\text{autopred}}$ is the lag1 autocorrelation of the residuals of the replicated datasets and τ_{sq} is the precision.

1.6) Gelman plot:



Shrink factor for every parameter converges to 1 immediately after specified number of iterations (10000). Shrink factor values for most parameter start from way below of 1.2.

Posterior Summary:

```

Iterations = 20001:30000
Thinning interval = 1
Number of chains = 3
Sample size per chain = 10000

```

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
beta0	9.403e+02	5.3091063	3.065e-02	3.044e-02
beta1	7.881e-03	0.0038419	2.218e-05	2.265e-05
beta2	3.993e+00	0.6200667	3.580e-03	3.580e-03
beta3	-2.553e+01	6.7302151	3.886e-02	3.866e-02
lag1auto	9.871e-02	0.0461396	2.664e-04	2.663e-04
lag1autopred	-1.649e-02	0.1264006	7.298e-04	7.267e-04
pppval	8.028e-01	0.3978912	2.297e-03	2.297e-03
tausq	6.042e-04	0.0001135	6.556e-07	7.019e-07

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta0	9.299e+02	9.368e+02	9.403e+02	943.841961	9.508e+02
beta1	2.818e-04	5.332e-03	7.889e-03	0.010445	1.545e-02
beta2	2.757e+00	3.583e+00	3.995e+00	4.404021	5.201e+00
beta3	-3.864e+01	-3.006e+01	-2.553e+01	-21.044060	-1.231e+01
lag1auto	1.209e-03	6.874e-02	1.021e-01	0.132057	1.789e-01
lag1autopred	-2.623e-01	-1.034e-01	-1.604e-02	0.070361	2.291e-01
pppval	0.000e+00	1.000e+00	1.000e+00	1.000000	1.000e+00
tausq	4.015e-04	5.239e-04	5.976e-04	0.000677	8.426e-04

DIC values:

```

> logit.dic.pd
Mean deviance: 619.4
penalty 4.072
Penalized deviance: 623.5
> logit.dic.popt
Mean deviance: 619.4
penalty 8.867
Penalized deviance: 628.3

```

Analysis:

Larger values of population density will cause small increase in the death rate since, the value of slope is positive and the credible sets for its slope does not have zero. Larger values of percentage of non white population will cause great increase in the death rate since, the value of slope is positive and far away from 0. The credible sets for its slope does not have zero. Larger values of percentage of number of years of education of adults will

cause great decrease in the death rate since, the value of slope is negative and far away from 0. The credible sets for its slope does not have zero. lag1autopred (lag 1 autocorrelation in the residuals from the replicated datasets) has a mean very close to 0 and 0 is near the center of 95% credible set. The lag1auto (lag 1 autocorrelation in the residuals from the real dataset) has a mean close to 0 but farther from 0 as compared to mean of lag1autopred and its credible set does not contain 0. Most importantly, predictive p-values is 0.8 approx. indicating that lag1auto was larger than lag1autopred in about 80% of the replications. Although value close to 0.5 is preferred but 0.8 is not extreme, the real data is not a typical of data generated from this model. A value greater than 0.9 would have suggested that we need to expand the model to account for autocorrelation in the data. The DIC values for this model is smaller than all the models we have discussed so far.

Overall Analysis after evaluation of three, two and one parameters models:

With the help of three parameters model, we actually confirm the results of analysis of one parameter models and two parameters models that percentage of non white population and number of years of education of adults are most important predictor variables for estimating death rate and population density alone may not be the good predictor but it becomes important when combined with other variables. That is why when we combine all these three variables in three parameters models we get better results especially in terms of DIC. Improvement in the values of DIC is significant enough to consider all these variables together. We cannot completely ignore or discard any one variable out. Each predictor variable is contributing to make the estimation of death rate better. However in terms of predictive checking the results are not that great. Models that gives more importance to numbers of years of education of adults variable yields best result for predictive checking, it gives us the notion that if we have to single out any one predictor variable then it would be number of years of education of adults.

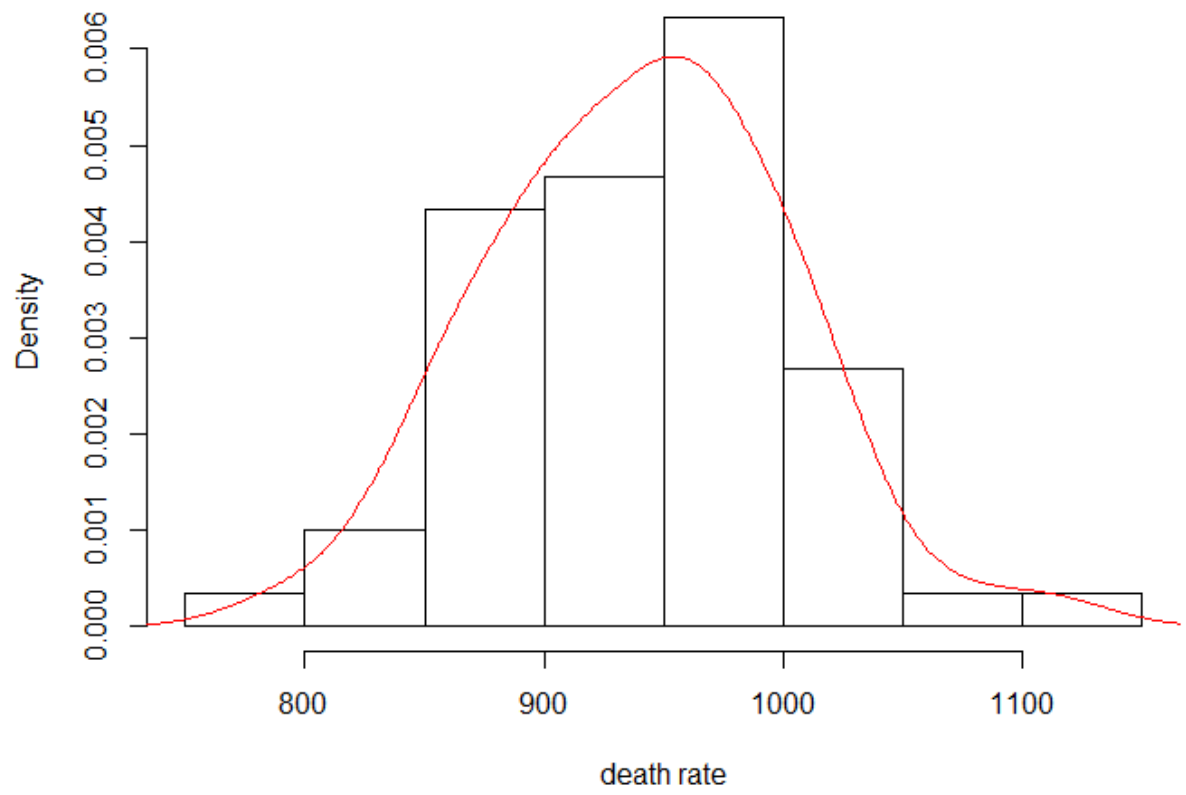
Appendix:

1)

```
hist(x = deathRateData$death, freq = FALSE, main="Histogram of the death rate data", xlab = "death rate", ylab = "Density")  
lines(x = density(deathRateData$death), col="red")
```

2)

Histogram of the death rate data



3)

```

deathRateCodeOne <- "
data {
  x_mean <- mean(x)
  for (i in 1:n) {
    xc[i] <- x[i] - x_mean
  }
}

model {
  for (i in 1:n) {
    y[i] ~ dnorm(mu[i], tausq)
    ypred[i] ~ dnorm(mu[i], tausq) # predicted values
    aresid[i] <- y[i] - mu[i] # actual residuals
    arespred[i] <- ypred[i] - mu[i] # predicted residuals
    mu[i] <- beta0 + beta1*xc[i]
  }

  # calculate lag 1 autocorrelation in residuals from real data
  mean1 <- mean(aresid[1:(n-1)])
  mean2 <- mean(aresid[2:n])

  for (i in 1:(n-1)) {
    summand[i] <- (aresid[i] - mean1) * (aresid[i+1] - mean2)
  }

  lag1auto <- sum(summand[]) / ((n-1)*sd(aresid[1:(n-1)])*sd(aresid[2:n]))

  # calculate lag 1 autocorrelation in residuals from replicated/predicted data
  mean1pred <- mean(arespred[1:(n-1)])
  mean2pred <- mean(arespred[2:n])

  for (i in 1:(n-1)) {
    summandpred[i] <- (arespred[i] - mean1pred) * (arespred[i+1] - mean2pred)
  }

  lag1autopred <- sum(summandpred[]) / ((n-1)*sd(arespred[1:(n-1)])*sd(arespred[2:n]))

  # is lag1auto > lag1autopred?
  pppval <- step(lag1auto - lag1autopred)

  beta0 ~ dnorm(0, 0.000001)
  beta1 ~ dnorm(0, 0.000001)
  tausq ~ dgamma(0.001, 0.001)
}

```

4)

```

## poulation density as a predictor variable

deathRateModelData <- list( x = deathRateData$popDens,
                             y = deathRateData$death, n = length(deathRateData$death))

## fit frequentist models to get ideas for initial values for JAGS

respmat <- cbind(deathRateModelData$y)
xcent <- deathRateModelData$x - mean(deathRateModelData$x)

summary(glm( respmat ~ xcent, family = gaussian(link="identity")))

```

5)

```

Call:
glm(formula = respmat ~ xcent, family = gaussian(link = "identity"))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-136.291  -38.894   -0.945   39.949  180.794

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.404e+02  7.809e+00 120.419  <2e-16 ***
xcent        1.136e-02  5.416e-03   2.097   0.0403 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 3658.87)

    Null deviance: 228308  on 59  degrees of freedom
Residual deviance: 212214  on 58  degrees of freedom
AIC: 666.53

Number of Fisher scoring iterations: 2

```

6)

```

## inits for logit model
deathRateModelInits <- list(list(beta0 = 908.8, beta1 = -0.0103, ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=350),
                             list(beta0 = 971.2, beta1 = 0.0329, ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=32542),
                             list(beta0 = 1002.4, beta1 = 0.0545, ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=5280))

deathRateModel <- jags.model(textConnection(deathRateCodeOne), data = deathRateModelData,
                             inits = deathRateModelInits, n.chains = 3, n.adapt = 0)

deathRateModelResults <- coda.samples(deathRateModel,
                                       c("beta0", "beta1", "tausq", "lag1auto", "lag1autopred", "pppval"),
                                       n.iter = 10000)

```

7)

```

deathRateModelData <- list( x = deathRateData$nonwh,
                             y = deathRateData$death, n = length(deathRateData$death))

## fit frequentist models to get ideas for initial values for JAGS

respmat <- cbind(deathRateModelData$y)
xcent <- deathRateModelData$x - mean(deathRateModelData$x)

summary(glm( respmat ~ xcent, family = gaussian(link="identity")))

```

8)


```

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-109.788   -32.738    -4.006    35.068    94.661

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  940.3584     6.1980 151.718 < 2e-16 ***
xcent         4.4896     0.7007   6.407 2.88e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 2304.948)

    Null deviance: 228308  on 59  degrees of freedom
Residual deviance: 133687  on 58  degrees of freedom
AIC: 638.81

Number of Fisher Scoring iterations: 2

```

9)

```

deathRateModelInits <- list(list(beta0 = 915.56, beta1 = 1.68, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=350),
                             list(beta0 = 965.16, beta1 = 7.28, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=32542),
                             list(beta0 = 989.9584, beta1 = 10.08, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=5280))

deathRateModel <- jags.model(textConnection(deathRateCodeOne), data = deathRateModelData,
                             inits = deathRateModelInits, n.chains = 3, n.adapt = 0)

deathRateModelResults <- coda.samples(deathRateModel,
                                       c("beta0", "beta1", "tausq", "lag1auto", "lag1autopred", "pppval"),
                                       n.iter = 10000)

```

10)

```

deathRateModelData <- list(x = deathRateData$educ,
                           y = deathRateData$death, n = length(deathRateData$death))

## fit frequentist models to get ideas for initial values for JAGS

respmat <- cbind(deathRateModelData$y)
xcent <- deathRateModelData$x - mean(deathRateModelData$x)

summary(glm( respmat ~ xcent, family = gaussian(link="identity")))

```

11)

```

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-151.708   -36.691    2.417    43.811   124.916

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  940.358      6.962  135.061 < 2e-16 ***
xcent        -37.604      8.306   -4.527 3.02e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 2908.544)

Null deviance: 228308  on 59  degrees of freedom
Residual deviance: 168696  on 58  degrees of freedom
AIC: 652.76

Number of Fisher Scoring iterations: 2

```

12)

```

deathRateModelInits <- list(list(beta0 = 912.67, beta1 = -70.8, ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=350),
                             list(beta0 = 968.03, beta1 = -4.4, ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=32542),
                             list(beta0 = 995.71, beta1 = 28.8, ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=5280))

deathRateModel <- jags.model(textConnection(deathRateCodeOne), data = deathRateModelData,
                             inits = deathRateModelInits, n.chains = 3, n.adapt = 0)

deathRateModelResults <- coda.samples(deathRateModel,
                                       c("beta0", "beta1", "tausq", "lag1auto", "lag1autopred", "pppval"),
                                       n.iter = 10000)

```

13)

```

deathRateCodeTwo <- "
  data {
    x1_mean <- mean(x1)
    x2_mean <- mean(x2)
    for (i in 1:n) {
      x1C[i] <- x1[i] - x1_mean
      x2C[i] <- x2[i] - x2_mean
    }
  }
  model {
    for (i in 1:n) {
      y[i] ~ dnorm(mu[i], tausq)
      ypred[i] ~ dnorm(mu[i], tausq) # predicted values
      aresid[i] <- y[i] - mu[i] # actual residuals
      arespred[i] <- ypred[i] - mu[i] # predicted residuals
      mu[i] <- beta0 + beta1*x1C[i] + beta2*x2C[i]
    }

    # calculate lag 1 autocorrelation in residuals from real data

    mean1 <- mean(aresid[1:(n-1)])
    mean2 <- mean(aresid[2:n])

    for (i in 1:(n-1)) {
      summand[i] <- (aresid[i] - mean1) * (aresid[i+1] - mean2)
    }

    lag1auto <- sum(summand[]) / ((n-1)*sd(aresid[1:(n-1)])*sd(aresid[2:n]))

    # calculate lag 1 autocorrelation in residuals from replicated/predicted data

    mean1pred <- mean(arespred[1:(n-1)])
    mean2pred <- mean(arespred[2:n])

    for (i in 1:(n-1)) {
      summandpred[i] <- (arespred[i] - mean1pred) * (arespred[i+1] - mean2pred)
    }

    lag1autopred <- sum(summandpred[]) / ((n-1)*sd(arespred[1:(n-1)])*sd(arespred[2:n]))

    # is lag1auto > lag1autopred?

    pppval <- step(lag1auto - lag1autopred)
    beta0 ~ dnorm(0, 0.000001)
    beta1 ~ dnorm(0, 0.000001)
    beta2 ~ dnorm(0, 0.000001)
    tausq ~ dgamma(0.001, 0.001)
  }
"

```

14)

```

deathRateModelData <- list( x1 = deathRateData$popDens, x2 = deathRateData$nonwh,
                             y = deathRateData$death, n = length(deathRateData$death))

## fit frequentist models to get ideas for initial values for JAGS

respmat <- cbind(deathRateModelData$y)
x1cent <- deathRateModelData$x1 - mean(deathRateModelData$x1)
x2cent <- deathRateModelData$x2 - mean(deathRateModelData$x2)

summary(glm( respmat ~ x1cent + x2cent, family = gaussian(link="identity")))

```

15)

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-96.182  -24.072    1.585   34.938   93.032

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.404e+02  5.853e+00 160.664 < 2e-16 ***
x1cent       1.151e-02  4.059e-03   2.836  0.00632 **
x2cent       4.500e+00  6.617e-01   6.800  6.84e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 2055.425)

Null deviance: 228308  on 59  degrees of freedom
Residual deviance: 117159  on 57  degrees of freedom
AIC: 632.89

Number of Fisher Scoring iterations: 2
```

16)

```
deathRateModelInits <- list(list(beta0 = 916.68, beta1 = -0.0005, beta2 = 1.86, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=350),
                             list(beta0 = 963.32, beta1 = 0.0235, beta2 = 7.14, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=32542),
                             list(beta0 = 986.64, beta1 = 0.0355, beta2 = 9.78, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=5280))

deathRateModel <- jags.model(textConnection(deathRateCodeTwo), data = deathRateModelData,
                             inits = deathRateModelInits, n.chains = 3, n.adapt = 0)

deathRateModelResults <- coda.samples(deathRateModel,
                                       c("beta0", "beta1", "beta2", "tausq", "lag1auto", "lag1autopred", "pppval"),
                                       n.iter = 10000)
```

17)

```
deathRateModelData <- list( x1 = deathRateData$popDens, x2 = deathRateData$educ,
                             y = deathRateData$death, n = length(deathRateData$death))

## fit frequentist models to get ideas for initial values for JAGS

respmat <- cbind(deathRateModelData$y)
x1cent <- deathRateModelData$x1 - mean(deathRateModelData$x1)
x2cent <- deathRateModelData$x2 - mean(deathRateModelData$x2)

summary(glm( respmat ~ x1cent + x2cent, family = gaussian(link="identity")))
```

18)

```

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-143.505   -36.227    2.502   39.485   132.850

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  940.358433   6.922248  135.846 < 2e-16 ***
x1cent       0.006408    0.004950   1.294 0.200720
x2cent      -34.915401    8.515340  -4.100 0.000132 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 2875.051)

Null deviance: 228308  on 59  degrees of freedom
Residual deviance: 163878  on 57  degrees of freedom
AIC: 653.02

Number of Fisher Scoring iterations: 2

```

19)

```

deathRateModelInits <- list(list(beta0 = 912.76, beta1 = -0.013192, beta2 = -68.92, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=350),
                             list(beta0 = 967.96, beta1 = 0.006408, beta2 = -0.92, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=32542),
                             list(beta0 = 995.56, beta1 = 0.026008, beta2 = 33.08, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=5280))

deathRateModel <- jags.model(textConnection(deathRateCodeTwo), data = deathRateModelData,
                             inits = deathRateModelInits, n.chains = 3, n.adapt = 0)

deathRateModelResults <- coda.samples(deathRateModel,
                                       c("beta0", "beta1", "beta2", "tausq", "lag1auto", "lag1autopred", "pppval"),
                                       n.iter = 10000)

```

20)

```

deathRateModelData <- list( x1 = deathRateData$nonwh, x2 = deathRateData$educ,
                             y = deathRateData$death, n = length(deathRateData$death))

## fit frequentist models to get ideas for initial values for JAGS

respmat <- cbind(deathRateModelData$y)
x1cent <- deathRateModelData$x1 - mean(deathRateModelData$x1)
x2cent <- deathRateModelData$x2 - mean(deathRateModelData$x2)

summary(glm( respmat ~ x1cent + x2cent, family = gaussian(link="identity")))

```

21)

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-103.857  -28.671    1.775    27.397    90.692

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  940.3584     5.4026 174.055 < 2e-16 ***
x1cent        3.9165     0.6245   6.271 5.16e-08 ***
x2cent       -28.9793     6.5904  -4.397 4.85e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 1751.314)

Null deviance: 228308  on 59  degrees of freedom
Residual deviance: 99825  on 57  degrees of freedom
AIC: 623.28

Number of Fisher Scoring iterations: 2
```

22)

```
deathRateModelInits <- list(list(beta0 = 918.8, beta1 = 1.42, beta2 = -55.33, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=350),
                             list(beta0 = 962.0, beta1 = 6.41, beta2 = -2.62, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=32542),
                             list(beta0 = 983.6, beta1 = 8.9, beta2 = 23.7, ypred=rnorm(60), .RNG.name="base::wichmann-Hill", .RNG.seed=5280))

deathRateModel <- jags.model(textConnection(deathRateCodeTwo), data = deathRateModelData,
                             inits = deathRateModelInits, n.chains = 3, n.adapt = 0)

deathRateModelResults <- coda.samples(deathRateModel,
                                       c("beta0", "beta1", "beta2", "tausq", "lag1auto", "lag1autopred", "pppval"),
                                       n.iter = 10000)
```

23)

```

deathRateCodeThree <- "
  data {
    x1_mean <- mean(x1)
    x2_mean <- mean(x2)
    x3_mean <- mean(x3)
    for (i in 1:n) {
      x1C[i] <- x1[i] - x1_mean
      x2C[i] <- x2[i] - x2_mean
      x3C[i] <- x3[i] - x3_mean
    }
  }
  model {
    for (i in 1:n) {
      y[i] ~ dnorm(mu[i], tausq)
      ypred[i] ~ dnorm(mu[i], tausq) # predicted values
      aresid[i] <- y[i] - mu[i] # actual residuals
      arespred[i] <- ypred[i] - mu[i] # predicted residuals
      mu[i] <- beta0 + beta1*x1C[i] + beta2*x2C[i] + beta3*x3C[i]
    }

    # calculate lag 1 autocorrelation in residuals from real data

    mean1 <- mean(aresid[1:(n-1)])
    mean2 <- mean(aresid[2:n])

    for (i in 1:(n-1)) {
      summand[i] <- (aresid[i] - mean1) * (aresid[i+1] - mean2)
    }

    lag1auto <- sum(summand[]) / ((n-1)*sd(aresid[1:(n-1)])*sd(aresid[2:n]))

    # calculate lag 1 autocorrelation in residuals from replicated/predicted data

    mean1pred <- mean(arespred[1:(n-1)])
    mean2pred <- mean(arespred[2:n])

    for (i in 1:(n-1)) {
      summandpred[i] <- (arespred[i] - mean1pred) * (arespred[i+1] - mean2pred)
    }

    lag1autopred <- sum(summandpred[]) / ((n-1)*sd(arespred[1:(n-1)])*sd(arespred[2:n]))

    # is lag1auto > lag1autopred?

    pppval <- step(lag1auto - lag1autopred)
    beta0 ~ dnorm(0, 0.000001)
    beta1 ~ dnorm(0, 0.000001)
    beta2 ~ dnorm(0, 0.000001)
    beta3 ~ dnorm(0, 0.000001)
    tausq ~ dgamma(0.001, 0.001)
  }
"

```

```
deathRateModelData <- list( x1 = deathRateData$popDens, x2 = deathRateData$nonwh, x3 = deathRateData$educ,
  y = deathRateData$death, n = length(deathRateData$death))

## fit frequentist models to get ideas for initial values for JAGS

respmat <- cbind(deathRateModelData$y)
x1cent <- deathRateModelData$x1 - mean(deathRateModelData$x1)
x2cent <- deathRateModelData$x2 - mean(deathRateModelData$x2)
x3cent <- deathRateModelData$x3 - mean(deathRateModelData$x3)

summary(glm( respmat ~ x1cent + x2cent + x3cent, family = gaussian(link="identity")))
```

25)

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-96.505  -17.806    3.825   21.688   88.036

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  940.358433    5.248891  179.154 < 2e-16 ***
x1cent         0.007877    0.003760    2.095 0.040723 *
x2cent         3.992259    0.607849    6.568 1.79e-08 ***
x3cent        -25.507930    6.613839   -3.857 0.000299 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 1653.051)

Null deviance: 228308  on 59  degrees of freedom
Residual deviance: 92571  on 56  degrees of freedom
AIC: 620.76

Number of Fisher Scoring iterations: 2
```

26)

```
deathRateModelInits <- list(list(beta0 = 919.4, beta1 = -0.0072, beta2 = 1.59, beta3 = -51.4,
  ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=350),
  list(beta0 = 961.32, beta1 = 0.0228, beta2 = 6.39, beta3 = 1.4,
  ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=32542),
  list(beta0 = 982.28, beta1 = 0.0378, beta2 = 8.79, beta3 = 27.8,
  ypred=rnorm(60), .RNG.name="base:wichmann-Hill", .RNG.seed=5280))

deathRateModel <- jags.model(textConnection(deathRateCodeThree), data = deathRateModelData,
  inits = deathRateModelInits, n.chains = 3, n.adapt = 0)

deathRateModelResults <- coda.samples(deathRateModel,
  c("beta0", "beta1", "beta2", "beta3", "tausq", "lag1auto", "lag1autopred", "pppval"),
  n.iter = 10000)
```