MID TERM Problem 1:

1. Let's try determining the type of disease based on the patient's Age. Use gradient descent (GD) to build your regression model (*model1*). Start by writing the GD algorithm and then implement it using R. [10 points]

→ Gradient Descent

At a theoretical level, gradient descent is an algorithm that minimizes functions i.e. cost. Given a function defined by a set of parameters, gradient descent starts with an initial set of parameter values and iteratively moves toward a set of parameter values that minimize the function. This iterative minimization is achieved using calculus, taking steps in the negative direction of the function gradient.

Cost function J(θ) is given below:

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2.$$

Here, we want to choose θ so as to minimize J(θ). Thus, the algorithm starts with some "initial guess" for θ, and that repeatedly changes θ to make J(θ) smaller, until hopefully we converge to a value of θ that minimizes J(θ).

$$\theta_j := \theta_j - \alpha\frac{\partial}{\partial\theta_j}J(\theta).$$

Here, α is called the learning rate and the algorithm takes a step in the direction of steepest decrease of J.

To implement the algorithm, take partial derivative of first equation, considering only one training sample.

$$\begin{aligned}\frac{\partial}{\partial\theta_j}J(\theta) &= \frac{\partial}{\partial\theta_j}\frac{1}{2}(h_\theta(x) - y)^2\\ &= 2\cdot\frac{1}{2}(h_\theta(x) - y)\cdot\frac{\partial}{\partial\theta_j}(h_\theta(x) - y)\\ &= (h_\theta(x) - y)\cdot\frac{\partial}{\partial\theta_j}\left(\sum_{i=0}^{n}\theta_i x_i - y\right)\\ &= (h_\theta(x) - y)\,x_j\end{aligned}$$

Thus, for all training set, equation would look like below:

$$\theta_j := \theta_j - \alpha(h_\theta(x^i) - y^i) * x^i_j$$

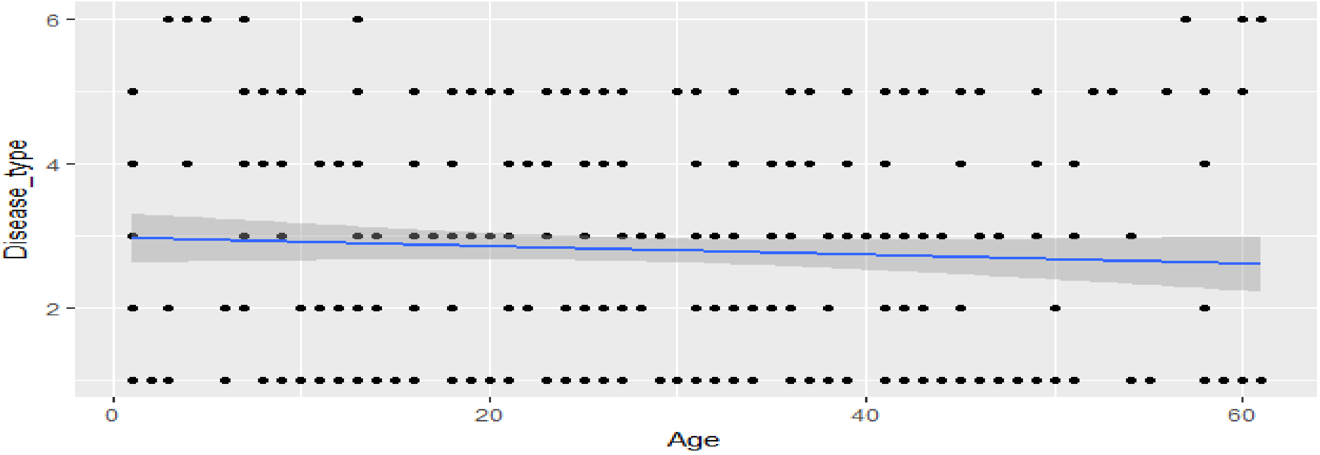
Repeat until convergence

```
{
        $\theta_j := \theta_j - \alpha(h_\theta(x^i) - y^i) * x^i_j$
}
```

This is the gradient descent algorithm to minimize the cost function.

Implementing gradient descent in R, lets first find the relationship between Age and Disease type using the linear regression:

```
> model <- lm(Disease_type~Age,derma_dta_GD)
> summary(model)
Call:
lm(formula = Disease_type ~ Age, data = derma_dta_GD)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.9698 -1.6992  0.1264  1.1806  3.3910 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.975805   0.177946  16.723   <2e-16 ***
Age         -0.006014   0.005478  -1.098    0.273    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.597 on 364 degrees of freedom
Multiple R-squared:  0.003301,	Adjusted R-squared:  0.0005624 
F-statistic: 1.205 on 1 and 364 DF,  . p-value: 0.273

> ggplot(derma_dta_GD, aes(x=Age, y=Disease_type)) + geom_point() +
+   stat_smooth(method="lm")
```



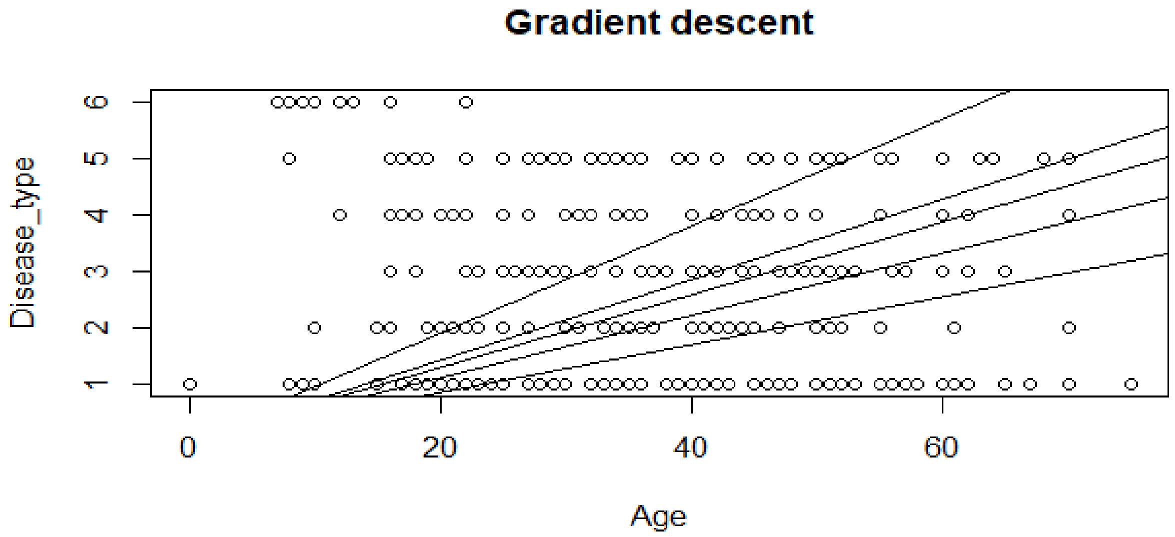
Gradient descent using the algorithm:

Please find the code in R file.

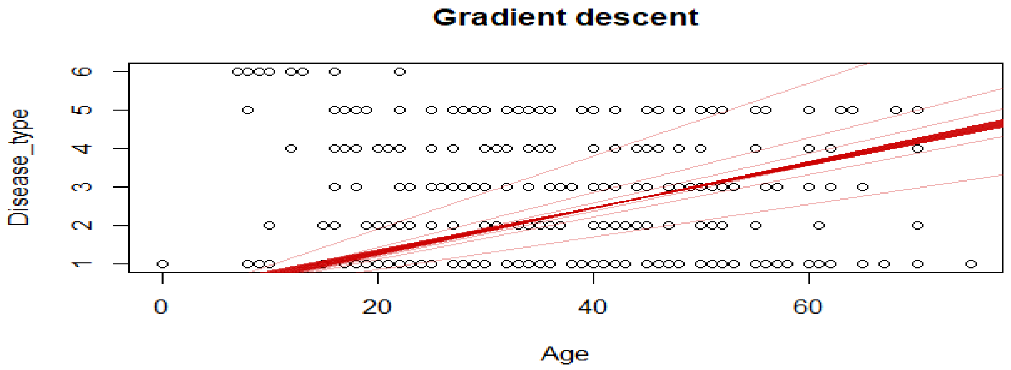
Plots are given below:

Theta values after the:

```
> print(theta)
          [,1]
[1,] 0.2730069
[2,] 0.0547229
> plot(Age,Disease_type, main = "Gradient descent")
> abline(coef = theta_history[[1]])
> abline(coef = theta_history[[2]])
> abline(coef = theta_history[[3]])
> abline(coef = theta_history[[4]])
> abline(coef = theta_history[[5]])
```

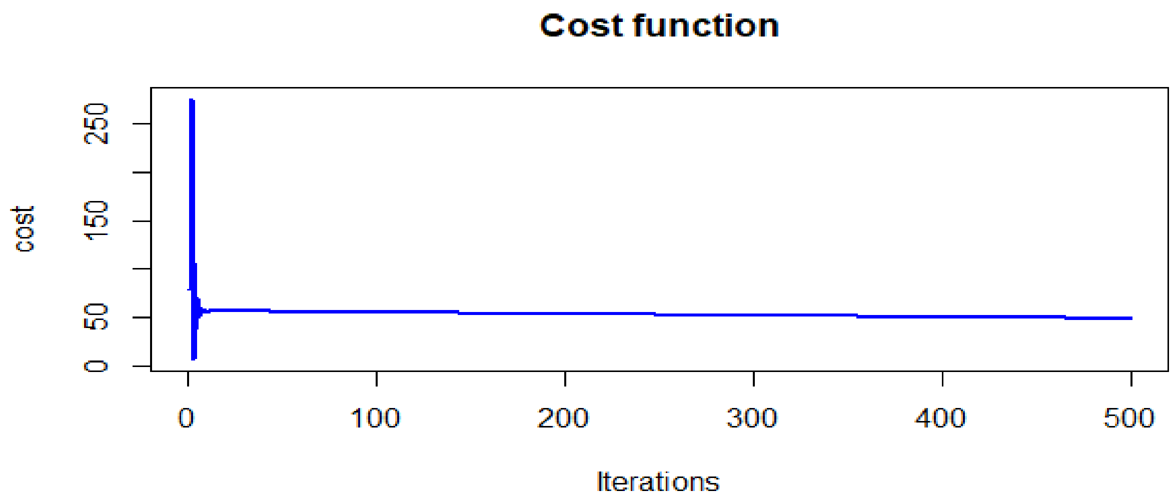


```
> plot(Age,Disease_type, main = "Gradient descent")
> for(i in c(1,2,3,4,5,seq(6,num_iterations, by = 10)))
+ {
+   abline(coef = theta_history[[i]], col=rgb(0.8,0,0,0.3))
+ }
```



```
> abline(coef = theta, col = 'blue')
> plot(cost_history, type = 'line', col = 'blue', lwd=2, main = 'Cost f
unction', ylab='cost', xlab = 'Iterations')
```

From the above plots, we can see that regression line is not covering the data points well as response variable Disease type is categorical.
Thus, gradient descent here is not very effective model to fit dermatology data.



2. Use random forest on the clinical as well as histopathological attributes to classify the disease type (*model2*). [5 points]

➔ Result from the Random Forest classification

```
> model <-randomForest(derma_train$Disease_type~ ., data = derma_train)
> model

Call:
randomForest(formula = derma_train$Disease_type ~ ., data = derma_train
)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 11

          Mean of squared residuals: 0.184276
                    % Var explained: 92.89
> prediction <- predict(model, newdata = derma_test)
> table(round(prediction), derma_test$Disease_type)

     1  2  3  4  5  6
  1 26  0  0  0  0  0
  2  0 11  0  0  0  0
  3  0  6 17  2  0  0
  4  0  0  1 13  0  0
  5  0  0  0  0 11  2
  6  0  0  0  0  0  3
> Accuracy = (26+11+17+13+11+3)/nrow(derma_test)
> Accuracy
[1] 0.8804348
```

From the Random forest classification results, we have achieved 88% of accuracy which is significantly good.

3. Use kNN on the clinical attributes and histopathological attributes to classify the disease type and report your accuracy (*model3*). [5 points]

➔ Result from the kNN classification:

```
> #KNN for k=3

> derma_pred <- knn(train = derma_dta.training, test = derma_dta.test,
cl = derma_dta.trainLabels, k=3)

> table(x=derma_pred, y=derma_dta.testLabels)
   y
x    1  2  3  4  5  6
  1 28  0  0  0  0  0
  2  0  8  1  2  0  0
  3  0  0 18  0  0  0
  4  0  6  0 12  0  0
  5  0  0  0  0 10  0
  6  0  1  0  0  0  5
> AccuracyKnn = (28+8+18+12+10+5) / nrow(derma_dta.test)
>
> AccuracyKnn
[1] 0.8901099
```

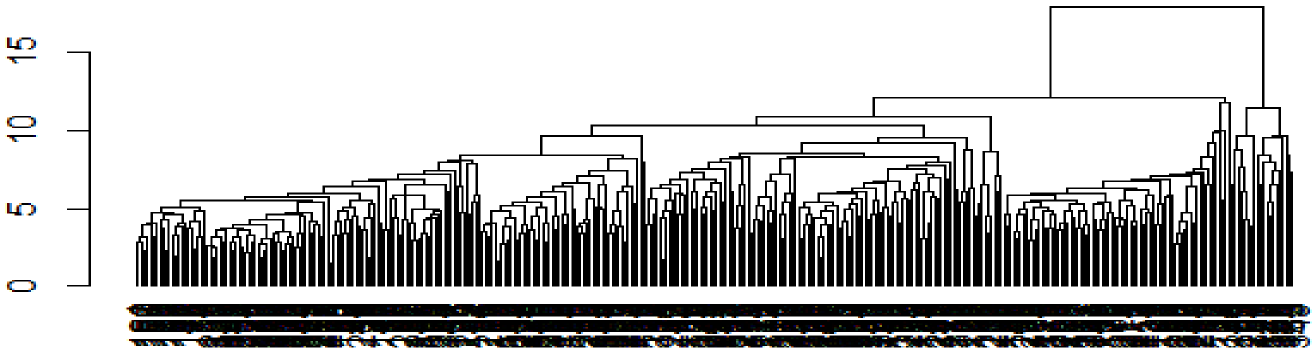
```
 > #kNN for k=4
 > derma_pred <- knn(train = derma_dta.training, test = derma_dta.test, cl = derma_d

 > table(x=derma_pred, y=derma_dta.testLabels)
    y
 x    1  2  3  4  5  6
   1 28  0  0  0  0  0
   2  0 11  1  4  0  0
   3  0  0 18  0  0  0
   4  0  3  0 10  0  0
   5  0  0  0  0 10  0
   6  0  1  0  0  0  5


 > AccuracyKnn = (28+11+18+10+10+5) / nrow(derma_dta.test)
 > AccuracyKnn
 [1] 0.9010989
```

Using the k nearest neighbor classification we have achieved 89% accuracy for k=3 and 90% accuracy for k=4. Thus, kNN is the good option for disease classification based on their accuracy.

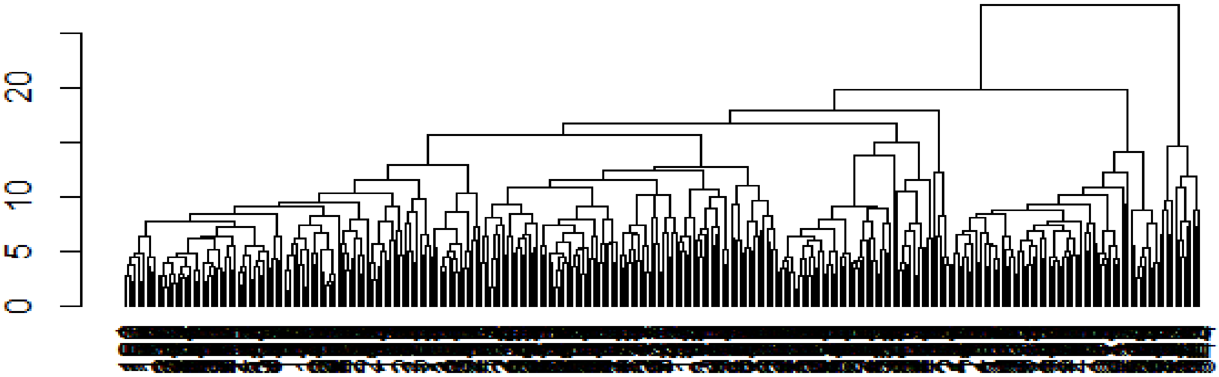
4. Finally, use at least two clustering algorithms and see how well these attributes can determine the disease type (*model4* and *model5*). [10 points]

➔ Agglomerative Clustering:

```
> response <- derma_dta[,35]
> predictor <- derma_dta[,1:34]
> clusters <- agnes(x=predictor, diss = FALSE, stand = TRUE, method = "
average")
> DendCluster <- as.dendrogram(clusters)
> plot(DendCluster)
```



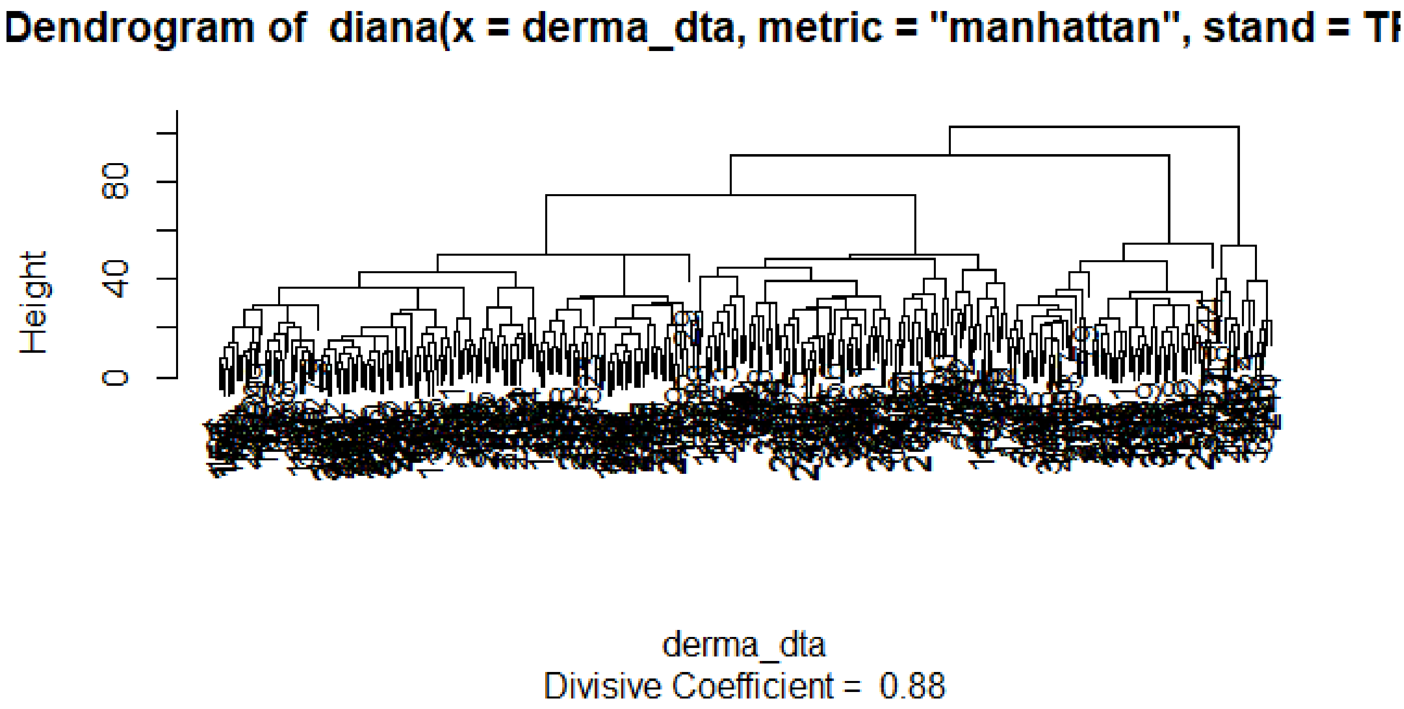
```
> clustersComplete <- agnes(x=predictor, diss = FALSE, stand = TRUE, method =
"complete")
> DendClusterComplete <- as.dendrogram(clustersComplete)
> plot(DendClusterComplete)
```



Agglomerative Clustering results i.e. dendrogram for dermatology data is not at all interpretable. Thus clustering is not a great choice for disease classification.

Divisive Clustering:

```
> fit <- diana(derma_dta, metric = "manhattan", stand = TRUE)
> plot(fit)
```



Model comparison:

1. Regression model using the Gradient Descent

→ We are using the dermatology department data to determine the type of disease based on their Age. In the data, age is continuous and disease type is categorical with 6 different disease types. Gradient descent works well on the continuous data; however, we can run the algorithm to check if the data fit the model. After implementing the gradient descent algorithm, we can see that it is not fitting the data well and is not an excellent choice to classify the dermatology data. However, we can use Softmax regression to predict the disease classes using the Age.

2. Random Forest

→ We ran random forest algorithm to classify the disease type using the all the available attributes i.e. clinical as well as histopathological attributes. As the data categorical, random forest worked well on the data. It explained 92.89% of variance in the data and has shown 88% of accuracy in prediction. It also taken care of the few missing values from the Age field. Thus, Random forest is the better choice to classify the dermatology data disease type.

3. kNN

→ We ran kNN algorithm to classify the disease type using the all the available attributes i.e. clinical as well as histopathological attributes. We ran it by setting different value for k and got better accuracy each time. For k=3, we got 89% accuracy and for k=4 it is 90%. Thus, kNN is also a better choice to classify the dermatology data disease type.

4. Clustering

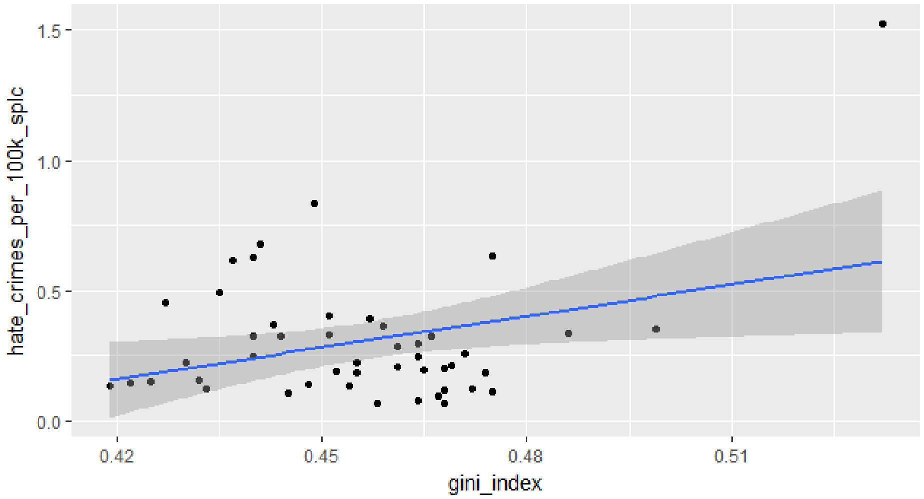
→ After running the agglomerative and divisive clustering on the dermatology data, the dendrogram results are not easy to interpret. Thus, clustering is not a great choice for to classify the dermatology data.

MID TERM Problem 2:

1. How does income inequality relate to the number of hate crimes and hate incidents? [5 points]

➔ We can perform the linear regression to identify the relation between the hate crime incidents and income inequality i.e. Gini index by state.

ggplot(hatecrime_dt, aes(x=gini_index, y=hate_crimes_per_100k_splc)) + geom_point() + stat_smooth(method="lm")

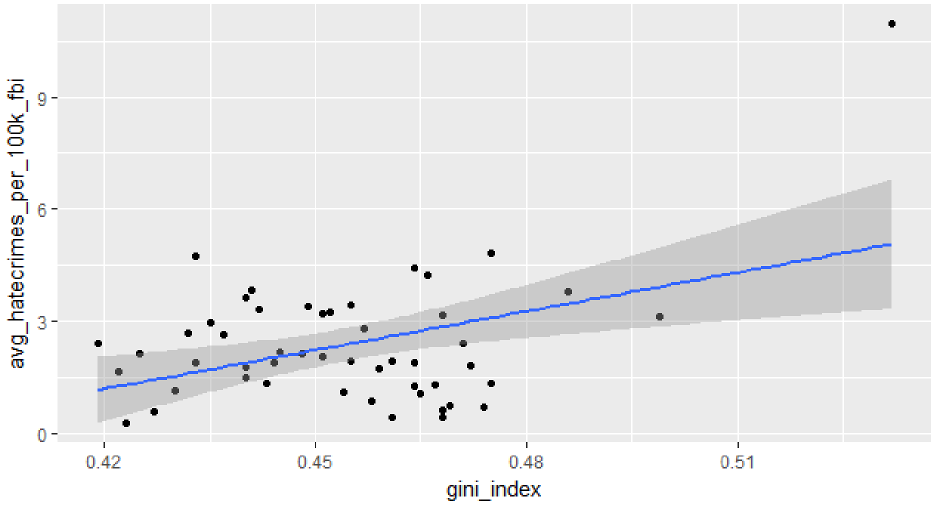


Linear Regression for gini_index and hate_crimes_per_100k_splc:

```
> model2 = lm(hate_crimes_per_100k_splc~gini_index, hatecrime_dt)
> summary(model2)
Call:
lm(formula = hate_crimes_per_100k_splc ~ gini_index, data = hatecrime_dt)

Residuals:
     Min       1Q   Median       3Q      Max
-0.28669 -0.14565 -0.04991  0.07356  0.91085
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -1.5275     0.7833  -1.950   0.0574 .
gini_index    4.0205     1.7177   2.341   0.0237 *
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2412 on 45 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.1085,	Adjusted R-squared:  0.08872
F-statistic: 5.478 on 1 and 45 DF,  p-value: 0.02374
```

The linear regression result shows there is very small p-value which can reject the NULL hypothesis and allows us to conclude there is relationship between hate crime and income in equality. However, R-squared value (0.1085) does not explain the variance well as it is close to 0, similarly F-statistic value is relatively less considering the number of data points. Thus, we can say income inequality has relationship with the latest hate crimes committed, but we cannot conclude the stronger relationship between them.

```
ggplot(hatecrime_dt, aes(x=gini_index, y=avg_hatecrimes_per_100k_fbi)) + geom_point() +
  stat_smooth(method="lm")
```



Linear Regression for gini_index and avg_hatecrimes_per_100k_fbi:

```
> model3 = lm(avg_hatecrimes_per_100k_fbi~gini_index, hatecrime_dt)
> summary(model3)
Call:
lm(formula = avg_hatecrimes_per_100k_fbi ~ gini_index, data = hatecrime_dt)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.4334 -0.9837 -0.1163  1.0005  5.8955 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)   
(Intercept)  -13.334      4.884  -2.730  0.00883 **
gini_index    34.571     10.743   3.218  0.00231 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.571 on 48 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared:  0.1775,	Adjusted R-squared:  0.1603 
F-statistic: 10.36 on 1 and 48 DF,  p-value: 0.002314
```

From the regression result, smaller P-value (0.00231) shows relationship between income inequality and average annual hate crime incidents. Income inequality is historically causing the hate crime incidents. However, the F-statistic (10.36) and R-squared (0.177) are not convincing the conclude the strong relationship.

Does historical hate crime (avg_hatecrimes_per_100k_fbi) show any relation to recent hate crime incidents along with gini index, let’s see the results below:

```
> model2 = lm(hate_crimes_per_100k_splc~gini_index+avg_hatecrimes_per_100k_fbi, hatecrime_dt)
> summary(model2)

Call:
lm(formula = hate_crimes_per_100k_splc ~ gini_index + avg_hatecrimes_per_100k_fbi,
   data = hatecrime_dt)
Residuals:
    Min      1Q  Median      3Q     Max
-0.46003 -0.11886  0.00065  0.08473  0.40705

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)             0.14725    0.59693  0.247    0.806
gini_index             -0.24218    1.34541  -0.180    0.858
avg_hatecrimes_per_100k_fbi  0.11408    0.01638    6.964 1.29e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1683 on 44 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.5759,	Adjusted R-squared:  0.5567
F-statistic: 29.88 on 2 and 44 DF,  p-value: 6.36e-09
```

Based on P-value (6.36e-09), we can see there is significant relationship between the historical hate crime and Gini index to recent hate crime incidents. Also, the R-squared value (0.57) explains good variance between the predictor and response variable. F-statistic values are relatively high compared to previous model to indicate the relationship between the predictor and response variable.

2. How can we predict the number of hate crimes and hate incidents from race/nature of the population? [5 points]
   - ➔ We can use linear regression to identify the relationship between the hate crime and race/nature of population.

```
> model = lm (hate_crimes_per_100k_splc~share_non_white+share_white_poverty+sha
re_non_citizen, hatecrime_dt)
> summary(model)

Call:
lm(formula = hate_crimes_per_100k_splc ~ share_non_white + share_white_poverty
+
    share_non_citizen, data = hatecrime_dt)

Residuals:
     Min       1Q   Median       3Q      Max
-0.36144 -0.14073 -0.01883  0.07267  1.06055

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)           0.4926     0.1950   2.526   0.0155 *
share_non_white      -0.2735     0.3763  -0.727   0.4715
share_white_poverty  -2.4658     1.6221  -1.520   0.1361
share_non_citizen     2.1822     1.8646   1.170   0.2486
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2444 on 41 degrees of freedom
  (6 observations deleted due to missingness)
Multiple R-squared:  0.1208,	Adjusted R-squared:  0.05651
F-statistic: 1.878 on 3 and 41 DF,  p-value: 0.1483
```

The p-value (0.148) from the above result shows that we can reject the NULL hypothesis to establish a relationship between response and predictor variables.

Thus, we can change the predictor variables to predict the hate crime rate.

```
> model = lm (hate_crimes_per_100k_splc~share_non_white+share_voters_vo
ted_trump+share_unemployed_seasonal, hatecrime_dt)
>
> summary(model)

Call:
lm(formula = hate_crimes_per_100k_splc ~ share_non_white + share_voters
_voted_trump +
    share_unemployed_seasonal, data = hatecrime_dt)

Residuals:
     Min       1Q   Median       3Q      Max
-0.39741 -0.08440 -0.01742  0.09551  0.54667

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)                 1.0770     0.1912   5.633 1.24e-06 ***
share_non_white            -0.6514     0.2349  -2.774  0.00817 **
share_voters_voted_trump   -1.7698     0.2619  -6.758 2.89e-08 ***
share_unemployed_seasonal   5.6685     3.2491   1.745  0.08819 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1814 on 43 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.5184,    Adjusted R-squared:  0.4848
F-statistic: 15.43 on 3 and 43 DF,  p-value: 5.889e-07
```

From the above regression result, the p-value (5.889e-07) for non-white people and people who voted to trump shows significant correlation for the hate crime incidents i.e. non-white people and trump voters have stronger relation as their corresponding p-values are significantly low. Also, R-squared value is showing 51.8% variance to the hate response variable. Their t values are away from the 0, which shows the relationship does exist.

Adding historical crime rate to see the impact on the existing model:

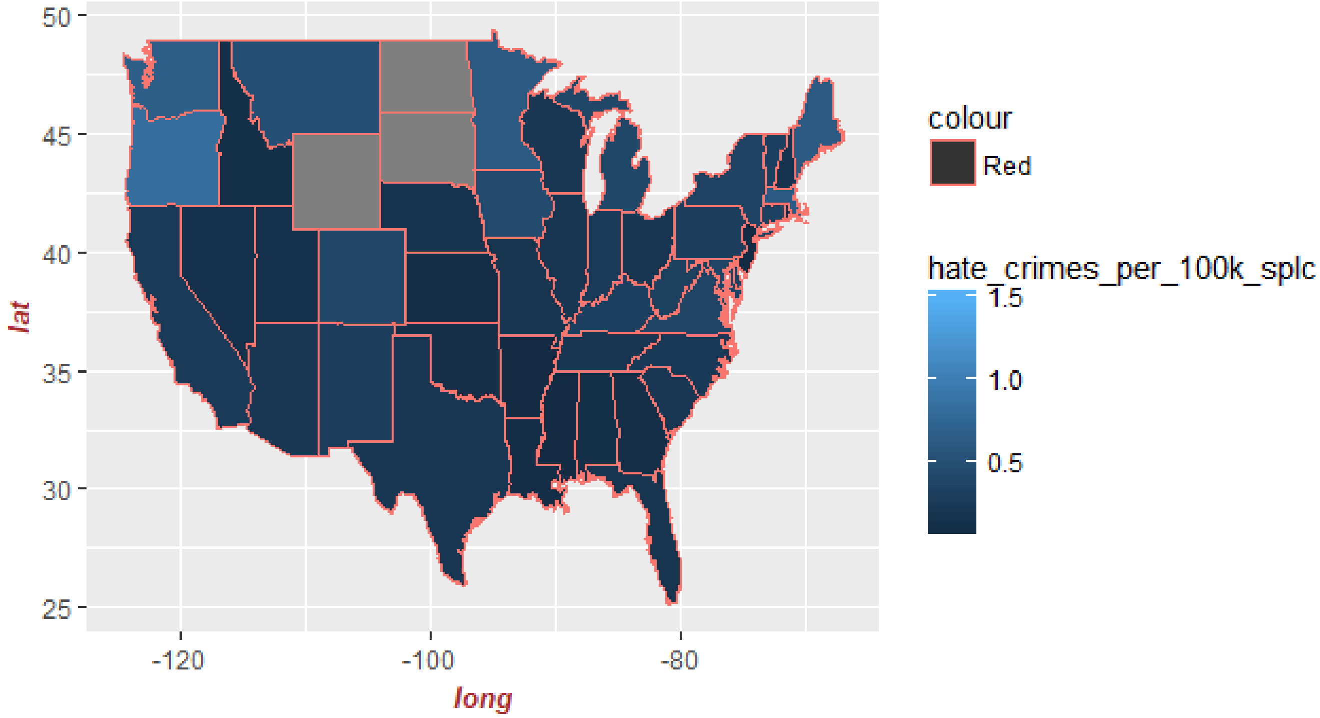
```
> model = lm (hate_crimes_per_100k_splc~share_non_white+share_voters_vo
ted_trump+share_unemployed_seasonal+avg_hatecrimes_per_100k_fbi, hatecr
ime_dt)
> summary(model)

Call:
lm(formula = hate_crimes_per_100k_splc ~ share_non_white + share_voters
_voted_trump +
    share_unemployed_seasonal + avg_hatecrimes_per_100k_fbi,
    data = hatecrime_dt)

Residuals:
     Min       1Q   Median       3Q      Max
-0.41115 -0.08739  0.00143  0.09036  0.32314

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)                   0.61177    0.18580   3.293 0.002019 **
share_non_white              -0.40346    0.19990  -2.018 0.049979 *
share_voters_voted_trump     -1.01150    0.26928  -3.756 0.000525 ***
share_unemployed_seasonal     2.55157    2.74849   0.928 0.358530
avg_hatecrimes_per_100k_fbi   0.07640    0.01634   4.674 3.04e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1489 on 42 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.6832,    Adjusted R-squared:  0.653
F-statistic: 22.64 on 4 and 42 DF,  p-value: 5.045e-10
```

If we add one more predictor historical average hate crime rate, it strongly supports the given model as the p-value is very low and R-square shows 68% of variation in the data. F-statistic value (22.64) is a Which show that if state has history of hate crime then recent hate crime rate for the particular state is high.

3. How does the number of hate crimes vary across states? Is there any similarity in number of hate incidents (per 100,000 people) between some states than in others — both according to the SPLC after the election and the FBI before it? [10 points]

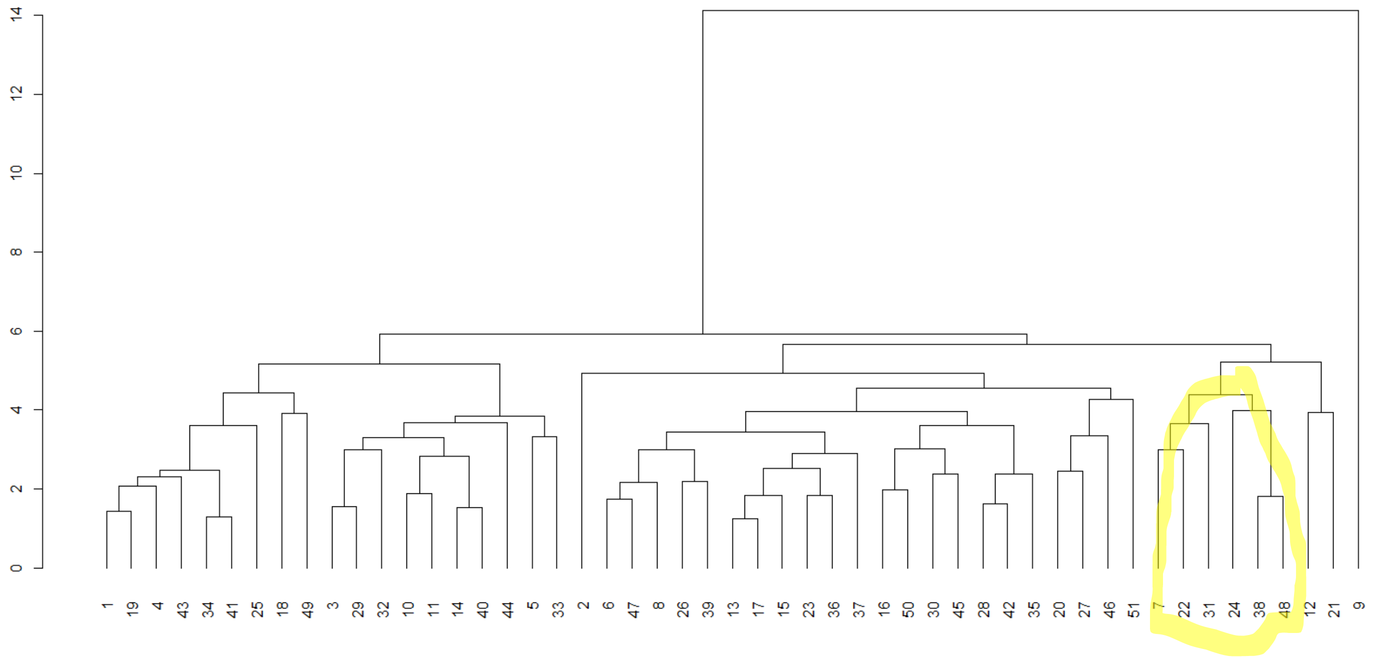
➔ Hate crime across the states is given in the below US heat map.



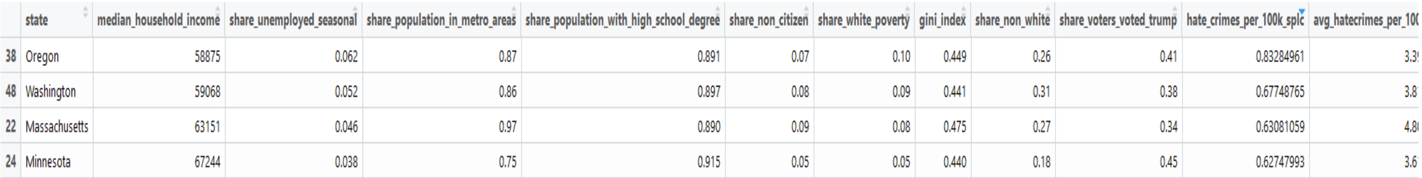
The lighter shade in color shows the high crime rate in the states after the 2016 elections and darker shades shows the low hate crime rate as shown in the scale

To find out the similarity between the number of hate crime incidents, we can perform the clustering to identify the similarity between the states considering the state as response variable.

```
> clusters <- agnes(x=predictor, diss = FALSE, stand = TRUE, method = "
average")
> DendCluster <- as.dendrogram(clusters)
> plot(DendCluster)
```



View(hatecrime_dt[c(31,22,24,48,38),])



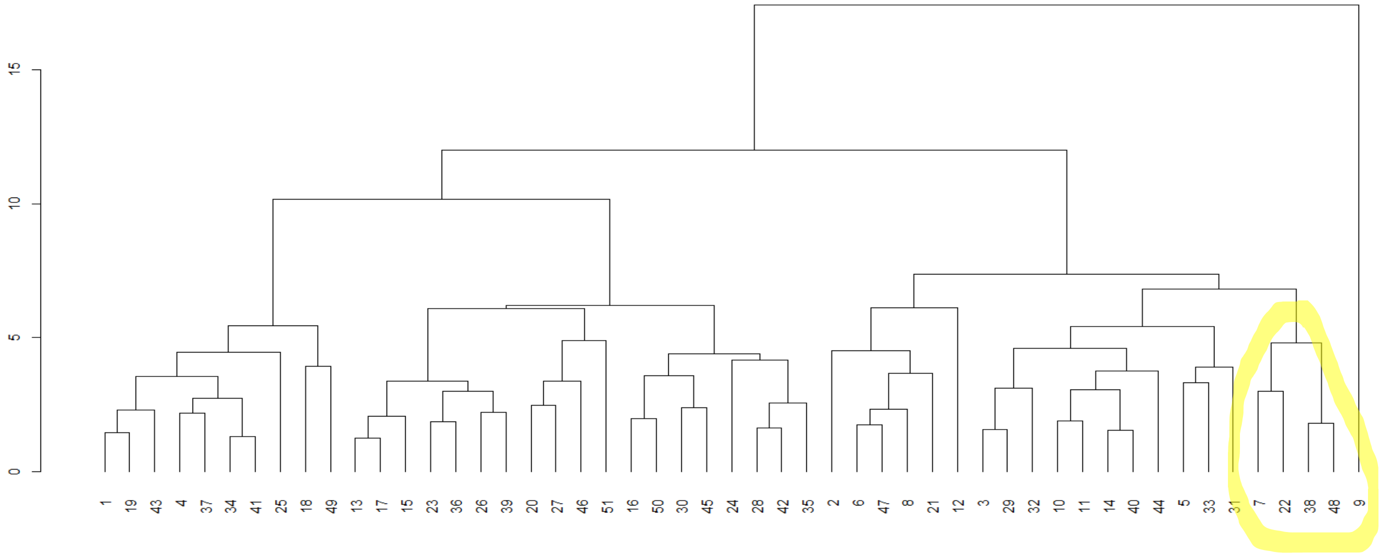
	state	median_household_income	share_unemployed_seasonal	share_population_in_metro_areas	share_population_with_high_school_degree	share_non_citizen	share_white_poverty	gini_index	share_non_white	share_voters_voted_trump	hate_crimes_per_100k_splc	avg_hatecrimes_per_10
38	Oregon	58875	0.062	0.87	0.891	0.07	0.10	0.449	0.26	0.41	0.83284961	3.3
48	Washington	59068	0.052	0.86	0.897	0.08	0.09	0.441	0.31	0.38	0.67748765	3.8
22	Massachusetts	63151	0.046	0.97	0.890	0.09	0.08	0.475	0.27	0.34	0.63081059	4.8
24	Minnesota	67244	0.038	0.75	0.915	0.05	0.05	0.440	0.18	0.45	0.62747993	3.6

Above histogram shows cluster of Oregon, Washington, Massachusetts and Minnesota, these states represent the highest crime rate during the 10 days after the election excluding the District of Columbia state which is top has top most crime rate after the election.

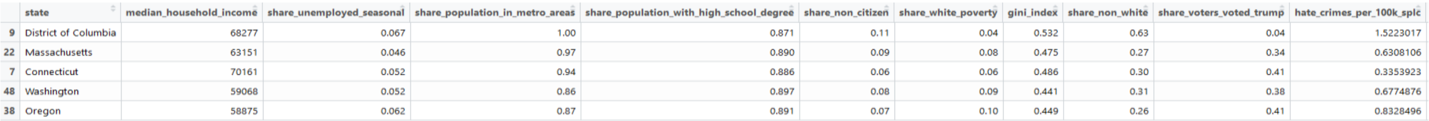
Also, we can see that the average hate crime is comparatively high for Oregon, Washington, Massachusetts and Minnesota states i.e., they have historical record of hate crime rate and has shown more crime rate during the 10 days after the election.

Clustering with method as complete

```
> clustersComplete <- agnes(x=predictor, diss = FALSE, stand = TRUE, method =
"complete")
> DendClusterComplete <- as.dendrogram(clustersComplete)
> plot(DendClusterComplete)
```



View(hatecrime_dt[c(9,22,7,48,38),])



	state	median_household_income	share_unemployed_seasonal	share_population_in_metro_areas	share_population_with_high_school_degree	share_non_citizen	share_white_poverty	gini_index	share_non_white	share_voters_voted_trump	hate_crimes_per_100k_splc
9	District of Columbia	68277	0.067	1.00	0.871	0.11	0.04	0.532	0.63	0.04	1.5223017
22	Massachusetts	63151	0.046	0.97	0.890	0.09	0.08	0.475	0.27	0.34	0.6308106
7	Connecticut	70161	0.052	0.94	0.886	0.06	0.06	0.486	0.30	0.41	0.3353923
48	Washington	59068	0.052	0.86	0.897	0.08	0.09	0.441	0.31	0.38	0.6774876
38	Oregon	58875	0.062	0.87	0.891	0.07	0.10	0.449	0.26	0.41	0.8328496

Above histogram shows cluster of Oregon, Washington, Connecticut and Massachusetts, these states represent the highest crime rate during the 10 days after the election excluding the District of Columbia state which is top has top most crime rate after the election.

Also, we can see that their average hate crime is comparatively high for Oregon, Connecticut and Massachusetts and Washington states i.e., they have historical record of hate crime rate and has shown more crime rate during the 10 days after the election.