PSTAT131 Final

Julissa Duenas

3/18/2021

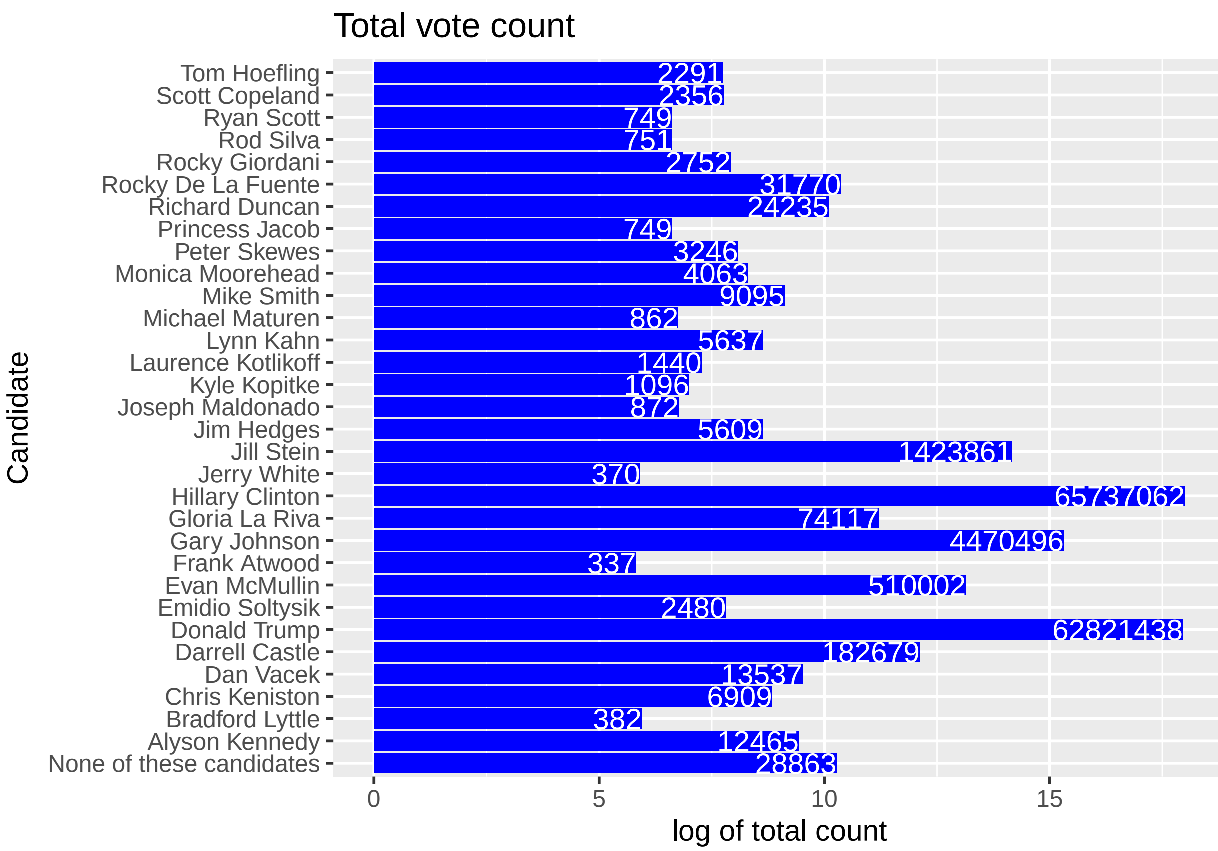
1. What makes voter behavior prediction (and thus election forecasting) a hard problem?
2. What was unique to Nate Silver's approach in 2012 that allowed him to achieve good predictions?
3. What went wrong in 2016? What do you think should be done to make future predictions better?

Question 4

answer question

Question 5

Question 6

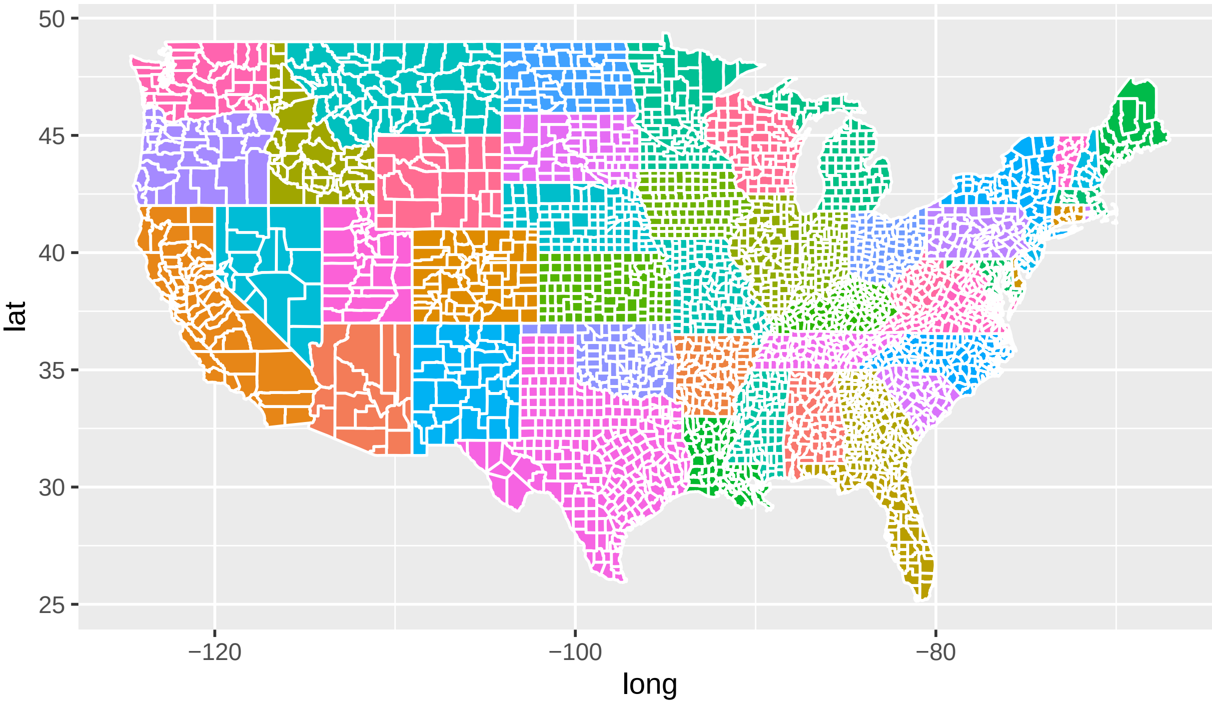


There are 32 candidates

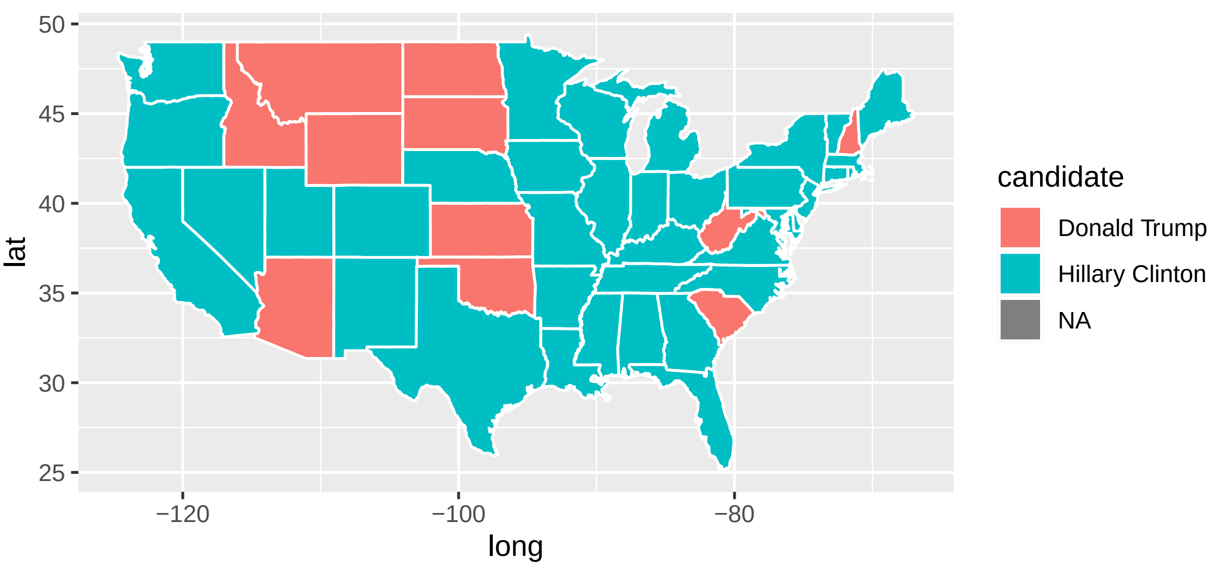
Question 7

Visualization

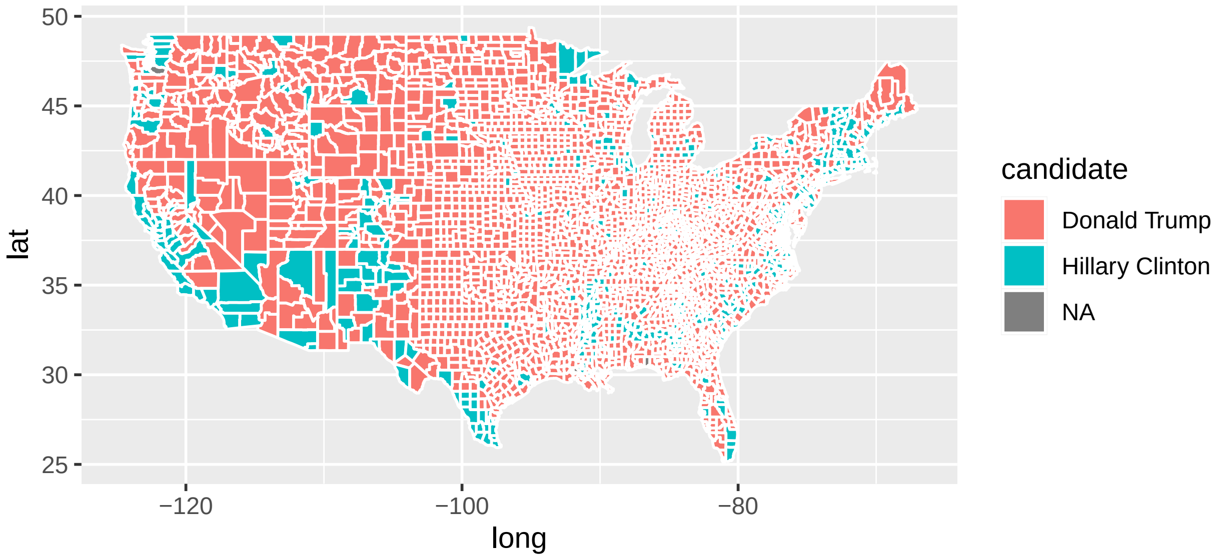
Question 8



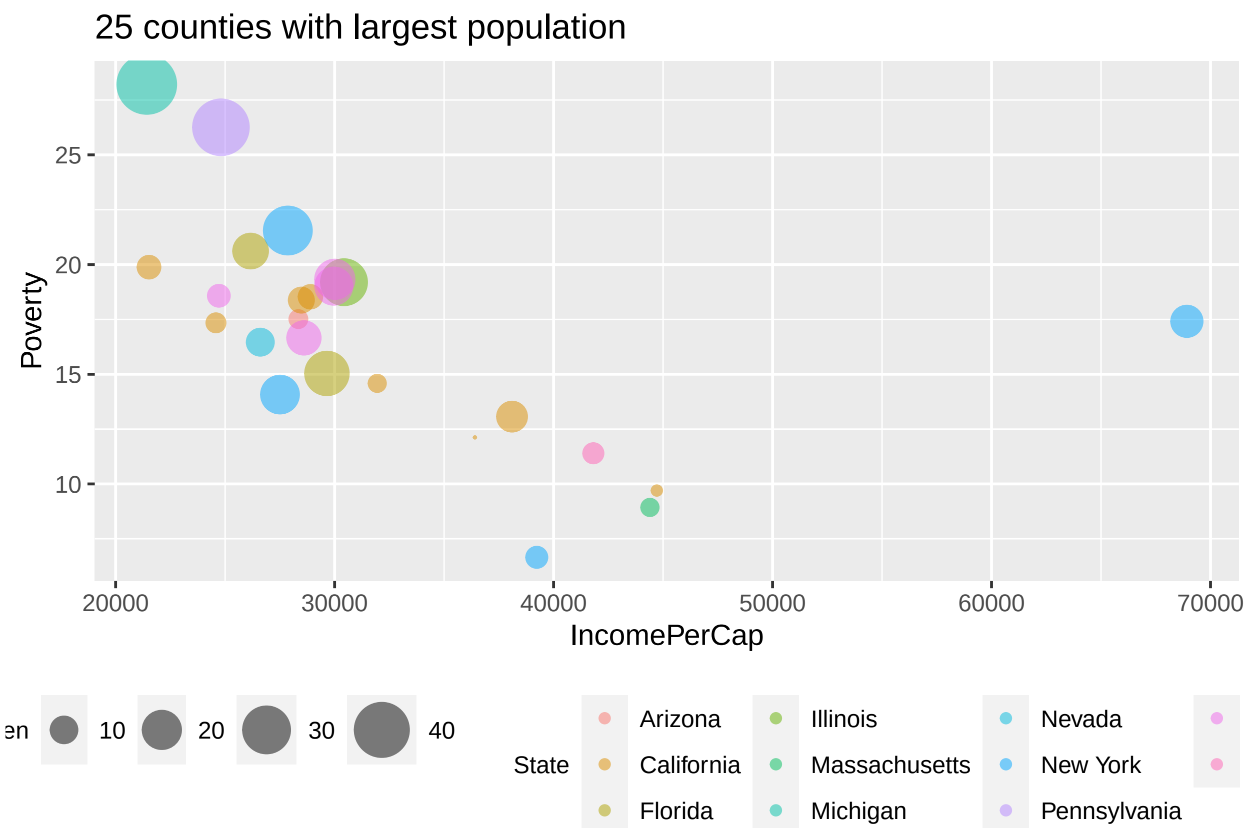
Question 9



Question 10



Question 11



Question 12

Dimensionality Reduction

Question 13

Table 1: County census data

State	County	Men	Women	White	Citizen	Income	IncomeErr	IncomePerCap	IncomePerC
Alabama	Autauga	48.43266	3348.805	75.78823	73.74912	51696.29	7771.009	24974.50	34
Alabama	Baldwin	48.84866	3934.167	83.10262	75.69406	51074.36	8745.050	27316.84	38
Alabama	Barbour	53.82816	1491.941	46.23159	76.91222	32959.30	6031.065	16824.22	24
Alabama	Bibb	53.41090	2930.106	74.49989	77.39781	38886.63	5662.358	18430.99	30
Alabama	Blount	49.40565	3562.081	87.85385	73.37550	46237.97	8695.786	20532.27	20
Alabama	Bullock	53.00618	1968.034	22.19918	75.45420	33292.69	9000.345	17579.57	31

Table 2: largest absolute values of PC1 for county

	PC1	PC2
IncomePerCap	-0.3524515	-0.1220681
ChildPoverty	0.3420583	-0.0081996
Poverty	0.3405654	-0.0143096

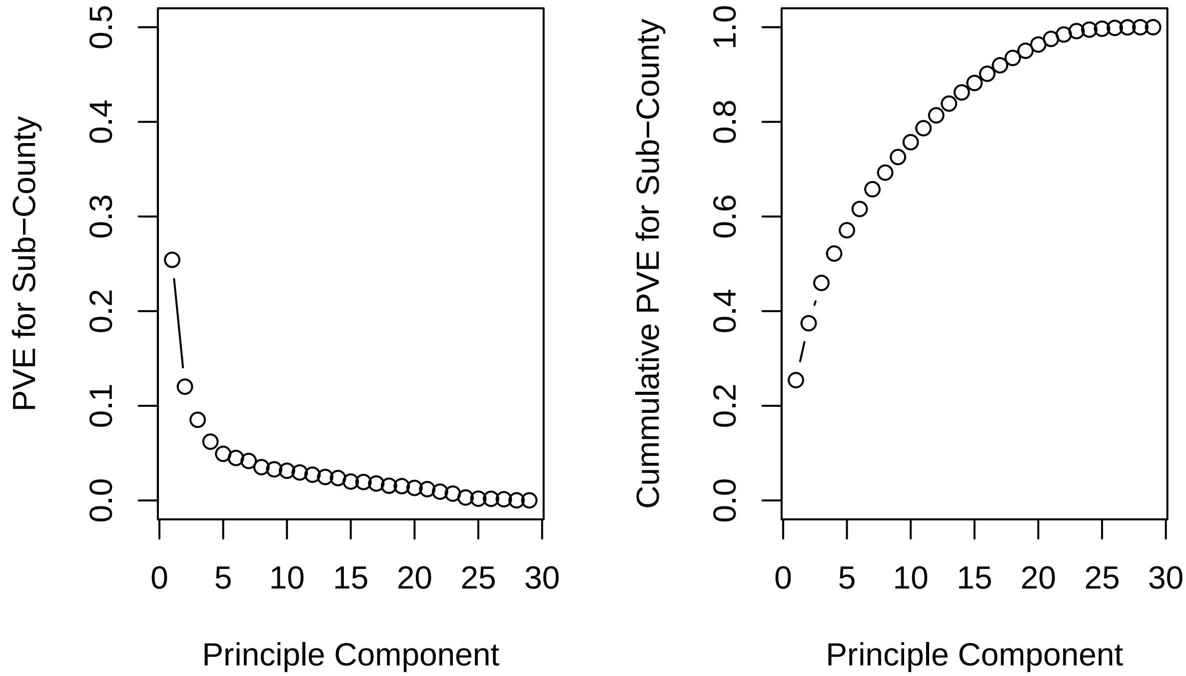
answer question

Question 14

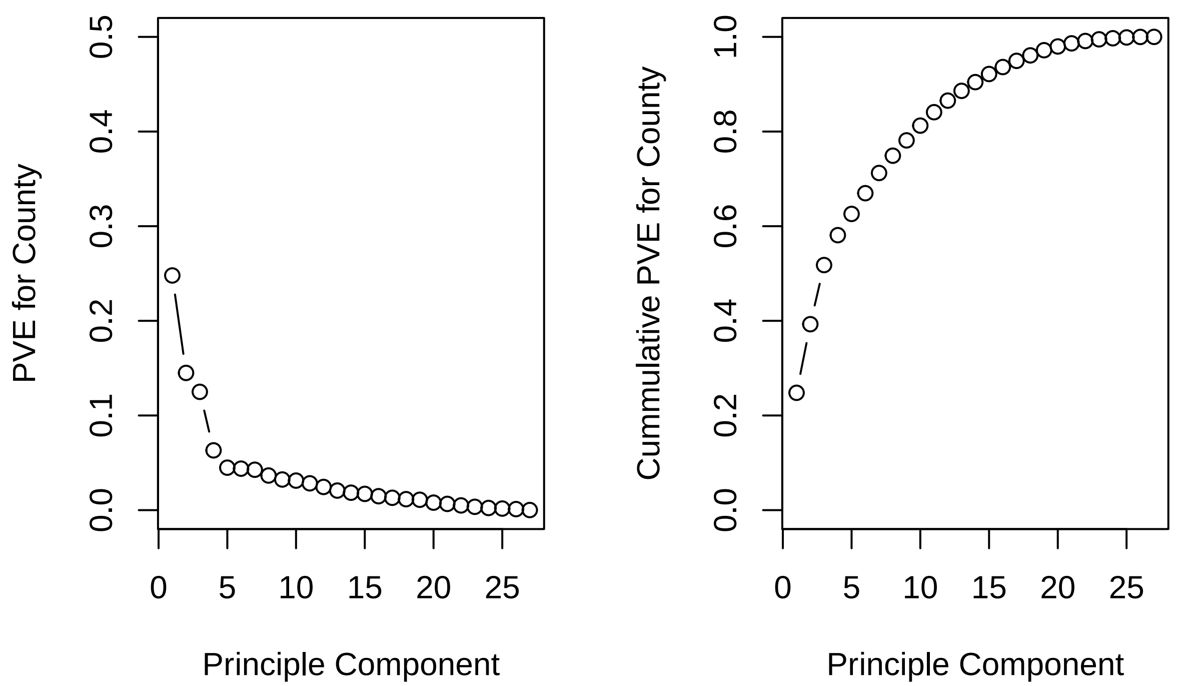
```
pr.subct.var <- subct.pca$sdev^2
pve.subct <- pr.subct.var/sum(pr.subct.var)
min.subct.pc <- min(which(cumsum(pve.subct)>=0.9))
#min.subct.pc #16
par(mfrow=c(1,2))
plot(pve.subct,xlab='Principle Component',ylab='PVE for Sub-County',type='b',ylim=c(0,0.5))
plot(cumsum(pve.subct),xlab='Principle Component',ylab='Cummulative PVE for Sub-County',ylim=c(0,1),typ
```

Table 3: largest absolute values of PC1 for sub-county

	PC1	PC2
IncomePerCap	0.3176826	-0.1660217
Professional	0.3062955	-0.1405477
Poverty	-0.3050684	-0.0494356




```
pr.ct.var <- ct.pca$sdev^2
pve.ct <- pr.ct.var/sum(pr.ct.var)
min.ct.pc <- min(which(cumsum(pve.ct)>=0.9))
#min.ct.pc #14
par(mfrow=c(1,2))
plot(pve.ct,xlab='Principle Component',ylab='PVE for County',type='b',ylim=c(0,0.5))
plot(cumsum(pve.ct),xlab='Principle Component',ylab='Cummulative PVE for County',ylim=c(0,1),type='b')
```

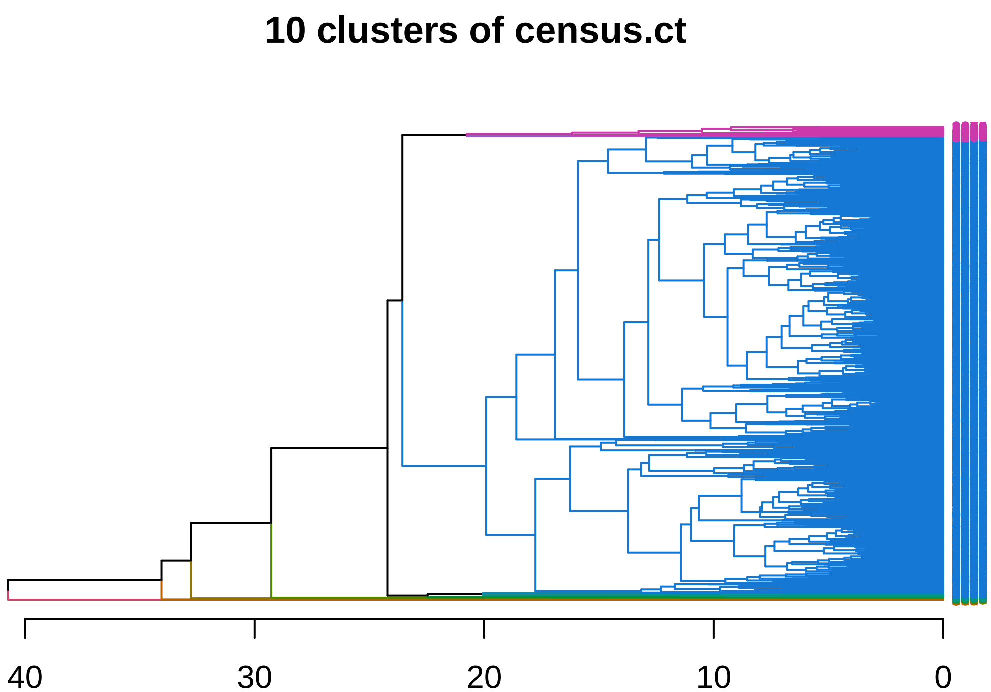


The minimum number of PCs to capture 90% of the variance is 14 for County and 16 for Sub-County

Clustering

Question 15

```
census.ct.scale <- as.data.frame(scale(census.ct[,-c(1,2)],center=TRUE,scale=TRUE))
census.ct.scale.dist <- dist(census.ct.scale,method='euclidean')
set.seed(1)
ct.hc <- hclust(census.ct.scale.dist,method = 'complete')
census.ct.dend <- as.dendrogram(ct.hc)
census.ct.dend=color_branches(census.ct.dend,k=10)
census.ct.dend=color_labels(census.ct.dend,k=10)
census.ct.dend=set(census.ct.dend,'labels_cex',0.5)
plot(census.ct.dend,horiz=TRUE,main='10 clusters of census.ct')
```




```
census.ct['Cluster'] <- cutree(ct.hc,10)
census.ct%>%filter(County=='San Mateo') #in cluster 5
```

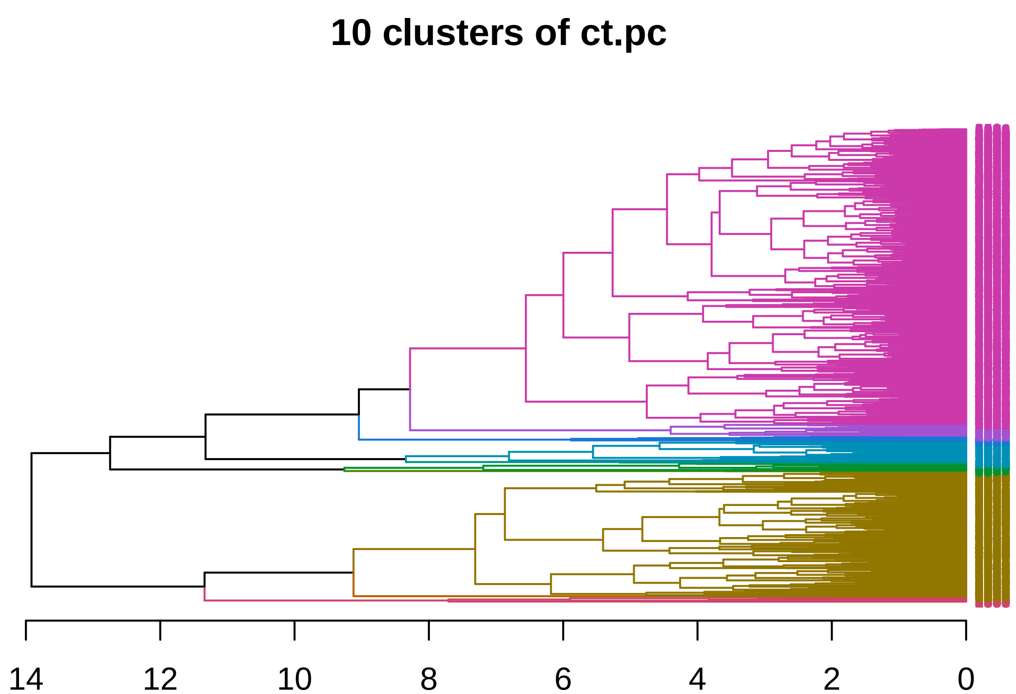
```
## # A tibble: 1 x 30
## # Groups:   State [1]
##   State County   Men Women White Citizen Income IncomeErr IncomePerCap
##   <chr> <chr>  <dbl> <dbl> <dbl>   <dbl>  <dbl>     <dbl>        <dbl>
## 1 Cali~ San M~  49.2 2757.  40.6    64.2 1.00e5    16123.       47881.
## # ... with 21 more variables: IncomePerCapErr <dbl>, Poverty <dbl>,
## #   ChildPoverty <dbl>, Professional <dbl>, Service <dbl>, Office <dbl>,
## #   Production <dbl>, Drive <dbl>, Carpool <dbl>, Transit <dbl>,
## #   OtherTransp <dbl>, WorkAtHome <dbl>, MeanCommute <dbl>, Employed <dbl>,
## #   PrivateWork <dbl>, SelfEmployed <dbl>, FamilyWork <dbl>,
## #   Unemployment <dbl>, Minority <dbl>, CountyTotal <dbl>, Cluster <int>
```

```
clusterct5 <- census.ct%>%filter(Cluster==5)
clusterct5
```

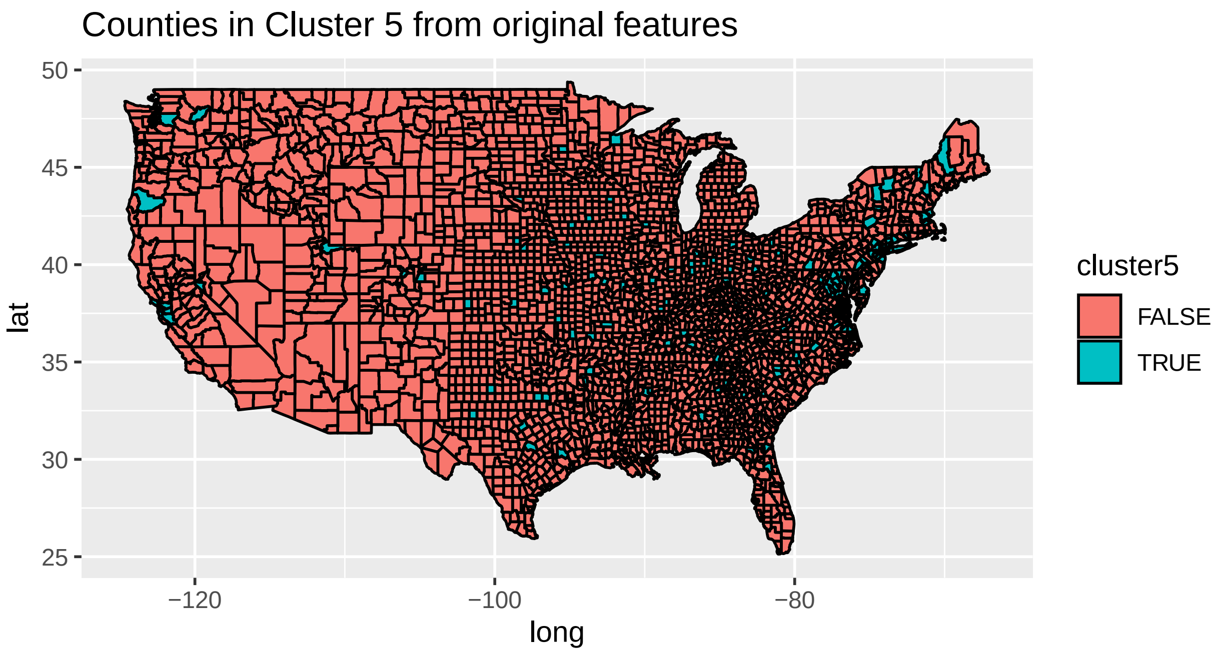
```
## # A tibble: 59 x 30
## # Groups:   State [19]
```

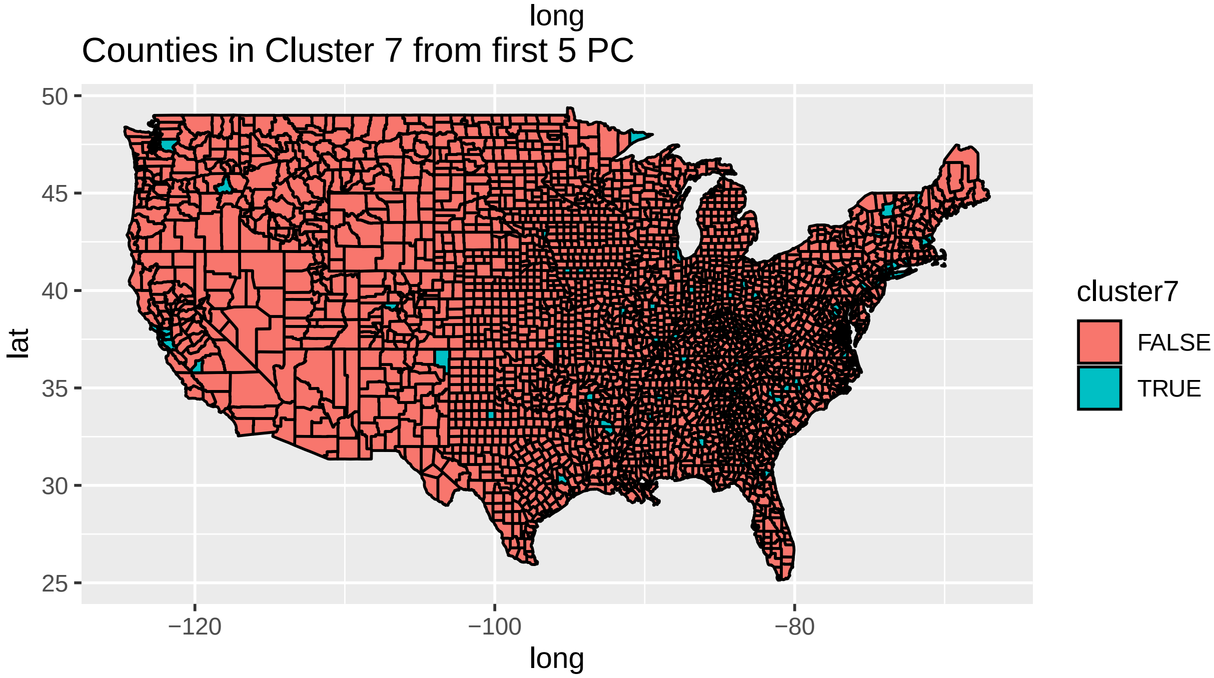
```
##    State County   Men Women White Citizen Income IncomeErr IncomePerCap
##    <chr> <chr>  <dbl> <dbl> <dbl>   <dbl>  <dbl>     <dbl>        <dbl>
##  1 Cali~ Alame~  49.0 2542.  33.0    64.7 8.31e4    12635.       37299.
##  2 Cali~ Contr~  48.8 3133.  45.8    65.6 8.96e4    13785.       39265.
##  3 Cali~ Marin   48.3 2764.  72.7    70.0 9.89e4    17538.       60993.
##  4 Cali~ San F~  50.9 2460.  41.3    73.6 8.54e4    14863.       52231.
##  5 Cali~ San M~  49.2 2757.  40.6    64.2 1.00e5    16123.       47881.
##  6 Cali~ Santa~  50.3 2771.  33.6    60.6 1.01e5    15215.       43880.
##  7 Colo~ Broom~  49.5 2282.  78.2    70.9 8.83e4    12724.       40135.
##  8 Colo~ Dougl~  49.6 2928.  84.0    68.2 1.07e5    12492.       45500.
##  9 Conn~ Fairf~  48.7 2582.  63.2    65.5 9.68e4    16315.       47742.
## 10 Dist~ Distr~  47.2 2180.  35.3    74.7 7.92e4    14309.       48504.
## # ... with 49 more rows, and 21 more variables: IncomePerCapErr <dbl>,
## #   Poverty <dbl>, ChildPoverty <dbl>, Professional <dbl>, Service <dbl>,
## #   Office <dbl>, Production <dbl>, Drive <dbl>, Carpool <dbl>, Transit <dbl>,
## #   OtherTransp <dbl>, WorkAtHome <dbl>, MeanCommute <dbl>, Employed <dbl>,
## #   PrivateWork <dbl>, SelfEmployed <dbl>, FamilyWork <dbl>,
## #   Unemployment <dbl>, Minority <dbl>, CountyTotal <dbl>, Cluster <int>
```

```
ct.pc.scale <- as.data.frame(scale(ct.pca$x[,1:5]),center=TRUE,scale=TRUE)
ct.pc.dist <- dist(ct.pc.scale,method='euclidean')
set.seed(1)
ct.pc.hc <- hclust(ct.pc.dist,method='complete')
ct.pc.dend <- as.dendrogram(ct.pc.hc)
ct.pc.dend=color_branches(ct.pc.dend,k=10)
ct.pc.dend=color_labels(ct.pc.dend,k=10)
ct.pc.dend=set(ct.pc.dend,'labels_cex',0.5)
plot(ct.pc.dend,horiz=TRUE,main='10 clusters of ct.pc')
```




```
census.ct['Cluster_PC'] <- cutree(ct.pc.hc,10)
#census.ct%>%filter(County=='San Mateo') #cluster 7
cluster7.pc <- census.ct%>%filter(Cluster_PC==7)
```

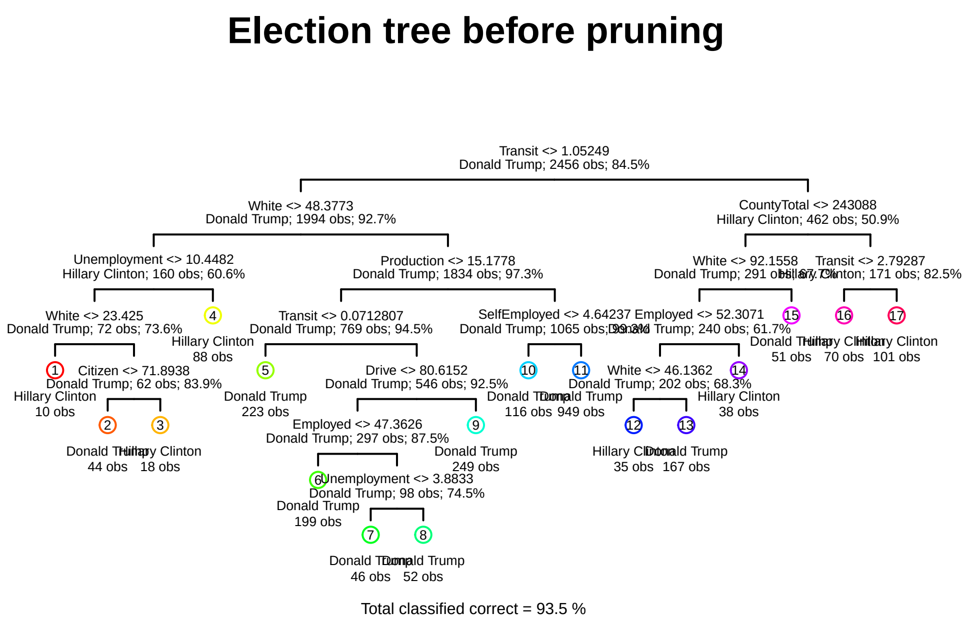




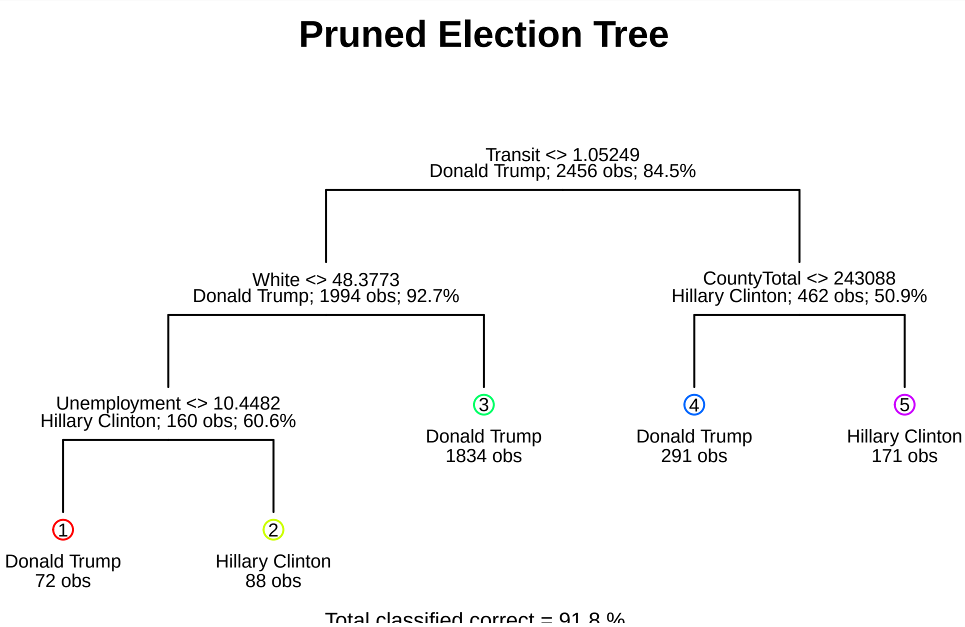
Classification

Question 16

```
election.tree <- tree(candidate~.,data=trn.cl)
draw.tree(election.tree,nodeinfo=TRUE,cex=0.45)
title('Election tree before pruning')
```




```
cv.election.tree <- cv.tree(election.tree,FUN=prune.misclass)
best.cv <- cv.election.tree$size[max(which(cv.election.tree$dev==min(cv.election.tree$dev)))]
#best.cv #8
pruned.election.tree <- prune.misclass(election.tree,best=best.cv)
draw.tree(pruned.election.tree,nodeinfo = TRUE,cex=0.55)
title('Pruned Election Tree')
```



#explain tree

Question 17