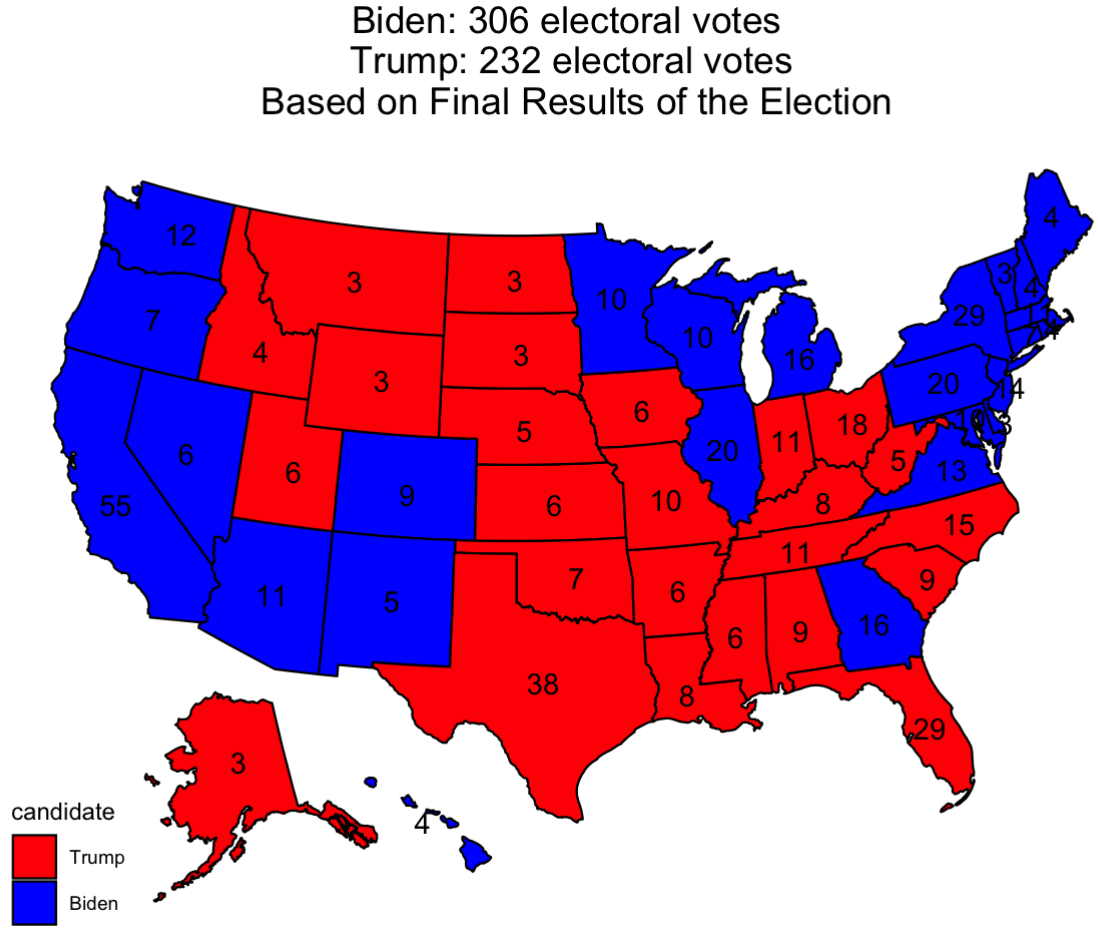
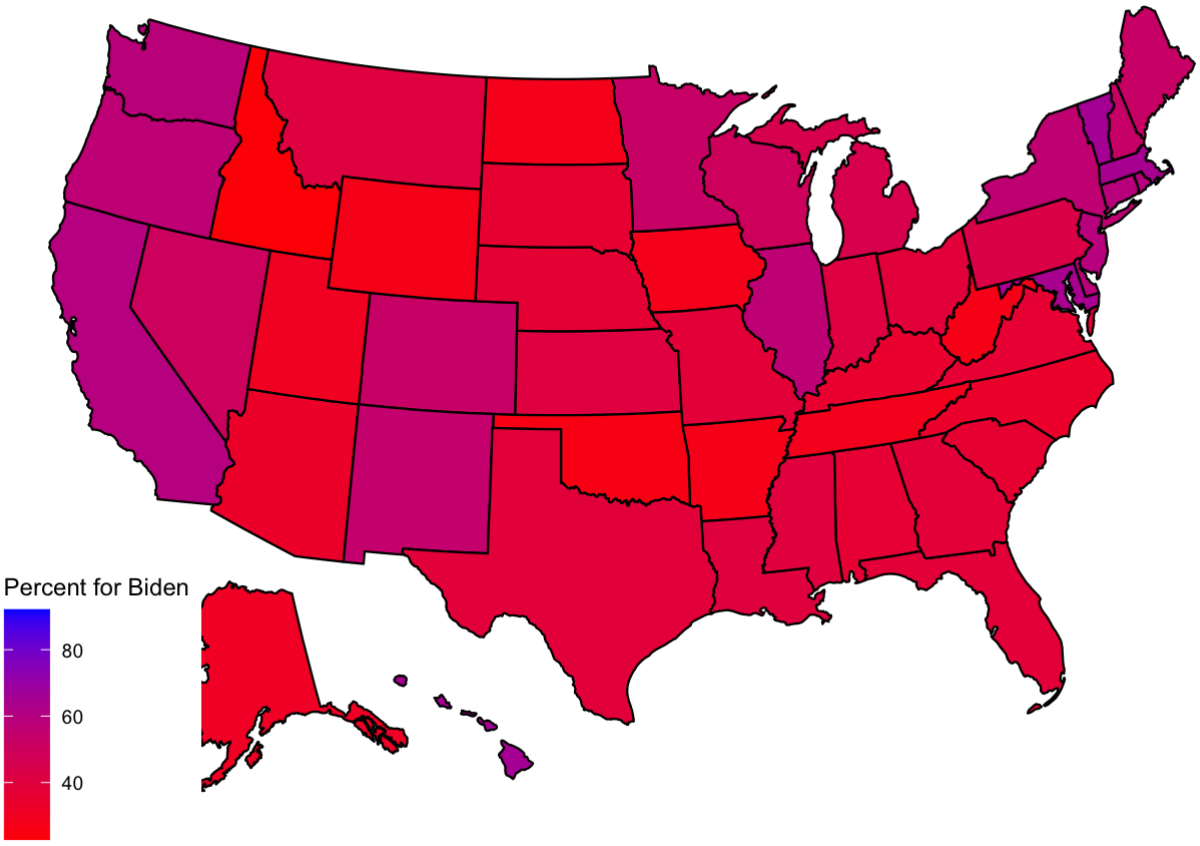
```
library(usmap)
library(dplyr)
library(tidyverse)
library(ggplot2)
library(lubridate)
library(gridExtra)
library(rgdal)

plot_election_results=function(electoral, vote_type, region) #defining function
{ #opens function
  #reading in data
  presidential <- read.csv("~/Downloads/USElection2020-NYT-Results-master/data/latest/presidential.csv", header=T)
  predidential1=subset(presidential, select=c(fips,name,state,votes,last_updated,absentee_votes,results_absentee_bidenj,results_bidenj,results_trumpd,results_absentee_trumpd))
  presidential1=as.data.frame(predidential1)
  attach(predidential1)
  electoralvotes<- read.csv("~/Downloads/electoralvotes.csv")
  attach(electoralvotes)
  #merging electoral votes file with presidential election data
  names(electoralvotes)[names(electoralvotes) == "state"] <- "State"
  joined_df <- merge(presidential1, electoralvotes, by.x = "state", by.y = "State", all.x = TRUE, all.y = FALSE)
  joined_df.1=group_by(joined_df,state)
  #getting state totals for total votes, total votes for each candidiate, total absantee votes, and total absantee votes for each candidate
  totaldf=summarise(joined_df.1,tot_votes=sum(votes),tot_abs=sum(absentee_votes),tot_biden=sum(results_bidenj),tot_trump=sum(results_trumpd),tot_abs_biden=sum(results_absentee_bidenj),tot_abs_trump=sum(results_absentee_trumpd))
  df2=cbind(electoralvotes,totaldf)
  #adding total abs votes for biden and trump to df
  df2=mutate(df2,in_person_totB=df2$tot_biden-df2$tot_abs_biden)
  df2=mutate(df2,in_person_totD=df2$tot_trump-df2$tot_abs_trump)
  #adding total in person votes
  df2=mutate(df2,in_person_tot=df2$tot_votes-df2$tot_abs)
  #renaming states so usmap can recorgnize
  state=c(state.name[1:8],"District of Columbia",state.name[9:50])
  dfF=df2[-c(1,3)]
  data=cbind(state,dfF)
  #getting coordinates for centers of each state
  state_centers=usmap_transform(tibble(state.center$x,state.center$y,state.name))
  data1=data[-9,] #taking out DOC bc usmap doesn't have it
  data1=mutate(data1,center_long=state_centers$state.center.x.1,center_lat=state_centers$state.center.y.1) #data for plotting electoral votes
  data1[2,12]=-1215632.238
  data1[2,13]=-1857070.429
  data1[11,12]=-408963.299
  data1[11,13]=-2117070.429
  #adding state abbreviations to fix when plotting specific region
  data1=mutate(data1, abb=c(state.abb[1:8],state.abb[9:50]))
  #calculating totalvote percentages for each candidate
  eleccalc=mutate(data1, perc_vote_B=(tot_biden/tot_votes)*100,perc_vote_T=(tot_trump/tot_votes)*100)
```

```
  #making subset of states where each candidate won the electoral votes
  electrump=eleccalc[eleccalc$tot_trump>eleccalc$tot_biden,]
  elecbiden=eleccalc[eleccalc$tot_biden>eleccalc$tot_trump,]
  #creating binary variable where electoral votes=1 if they go to biden and 0 if they go
to trump
  elecbiden1=mutate(elecbiden,elec=1)
  electrump1=mutate(electrump,elec=0)
  elec=rbind(elecbiden1,electrump1)
  #changing binary variable to factor so I can later change the scale from continuous to
discrete
  elec=mutate(elec,elec1=as.factor(elec))
  if(electoral==T)
  { #opens when we want to calculate electoral votes
    if(is.null(region)){ #assigning all state abbreviations to region if region is set t
o null and plotting when region is NULL
      region=data1$abb
      data2 = subset(elec, abb %in% region)
      plot_usmap(data=data2, values="elec1", regions="states",include=region)+
        scale_fill_manual(values = c(`0` = "red", `1` = "blue"), name = "candidate",labe
ls=c("Trump","Biden"))+ #changes legend to discrete and customize
        geom_text(data=data2,aes(x=center_long,y=center_lat,label=number.of.votes))+
        ggtitle(paste("Biden: 306 electoral votes \n Trump: 232 electoral votes \n Based
on Final Results of the Election"))+
        theme(plot.title = element_text(hjust=.5,size=14)) #adjusts position of title to
middle
    }
    else {
    #subsetting data specified by region
    data2 = subset(elec, abb %in% region)
    #calculating total electoral votes of specified region for each candidate
      biden_elec_tot=sum(data2$number.of.votes[data2$elec==1])
      trump_elec_tot=sum(data2$number.of.votes[data2$elec==0])
      plot_usmap(data=data2, values="elec1", regions="states",include=region)+
        scale_fill_manual(values = c(`0` = "red", `1` = "blue"), name = "candidate",labe
ls=c("Trump","Biden"))+ #changes legend to discrete and customize
        geom_text(data=data2,aes(x=center_long,y=center_lat,label=number.of.votes))+
        ggtitle(paste("Biden:", biden_elec_tot, "electoral votes \n Trump:", trump_elec_
tot, " electoral votes \n Based on Final Results of the Election"))+
        theme(plot.title = element_text(hjust=.5,size=14)) #adjusts position of title to
middle
   }
  } else if(electoral==F)
  { #opens else if electoral =F (popular vote)
    if(vote_type=="total")
    {
      plot_usmap(data=elec,values="perc_vote_B",region="state",include=region)+
        scale_fill_gradient(low = "red", high = "blue",name="Percent for Biden")+ #chang
es color scale of legend to reflect party/candidacy
        ggtitle(paste("Total Popular Vote"))+
        theme(plot.title = element_text(hjust=.5,size=14)) #adjusts position of title to
the middle of the screen and size of title
    } else if(vote_type=="absentee")
    {
      dataA=mutate(data,perc_abs_vote_B=(tot_abs_biden/tot_abs)*100) #calculating percen
```

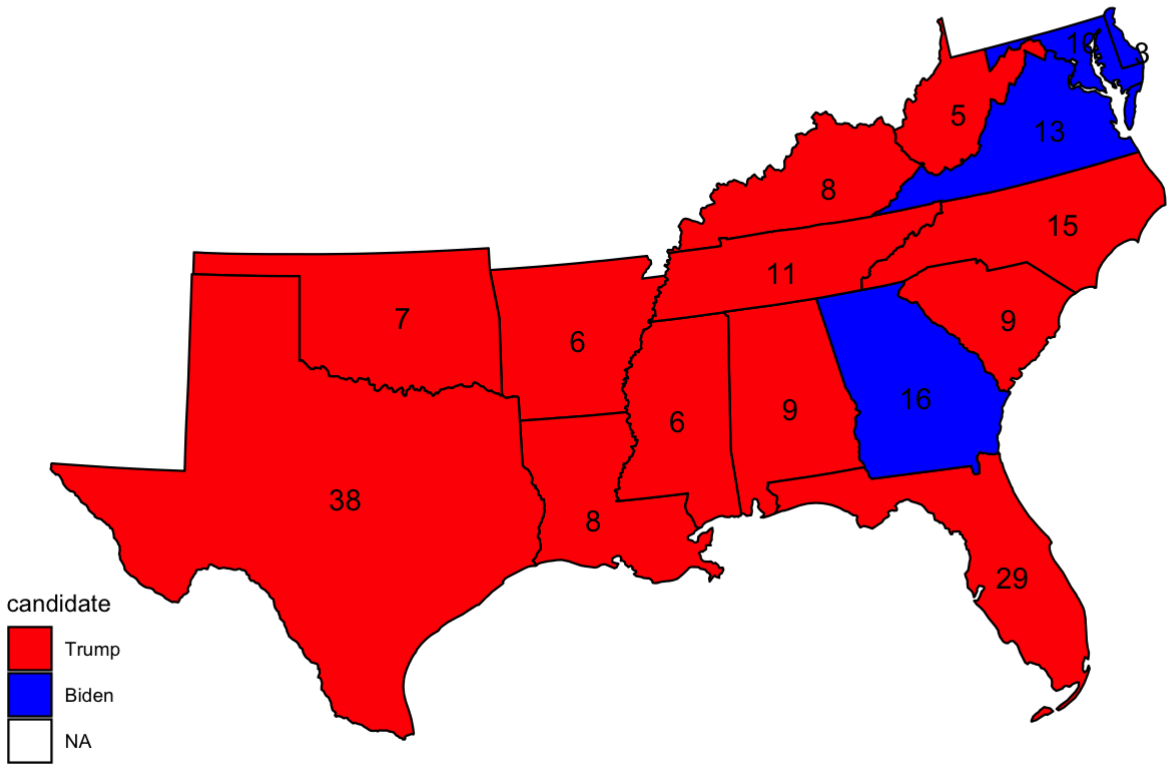
```
t absentee votes for biden
      plot_usmap(data=dataA,values="perc_abs_vote_B",region="state",include=region)+
        scale_fill_gradient(low = "red", high = "blue",name="Percent for Biden")+
        ggtitle(paste("Absentee Popular Vote"))+
        theme(plot.title = element_text(hjust=.5,size=14))
    } else if(vote_type=="in-person")
    {
      dataA=mutate(data,perc_inp_vote_B=(in_person_totB/(tot_votes-tot_abs))*100) #calcu
lating percent in person votes for biden
      plot_usmap(data=dataA,values="perc_inp_vote_B",region="state",include=region)+
        scale_fill_gradient(low = "red", high = "blue",name="Percent for Biden")+
        ggtitle(paste("In-Person Popular Vote"))+
        theme(plot.title = element_text(hjust=.5,size=14))
    }else #error message for invalid vote type
    {
      print("Error: invalid argument for 'vote_type'. Valid arguements include: 'in-pers
on','absentee',or'total'")
    }
  } #closes  for popular vote
  else if (electoral!=F & electoral!=T) #error message for invalid electoral input
  {
    print("Error: invalid argument for 'electoral'. Input must be either T (true) or F
 (false)")
  }

} #closes function
plot_election_results(electoral=T,vote_type="total",region=NULL)
```




```
plot_election_results(electoral=F,vote_type="in-person",region=NULL)
```




```
plot_election_results(electoral=T, vote_type="in-person", region=.south_region)
```




```
plot_election_results(electoral=F,vote_type="in-person",region=.south_region)
```

In-Person Popular Vote