A Retrospective Exploration of the US Senate through Data

Jordan Poles, Tony Chen and Aparna Narendrula

November 29, 2015

An Introduction to the Data

Our analysis covers a dataset which has been extracted from several different sources covering the timespan from the 101st to 113th US Congress. This corresponds to years of 1989 through 2014. The goal of our analysis was to assess trends in the members and bills passed by the US Senate. Our group found it most sensible to focus our efforts on one of the two chambers of Congress (the other being the House of Representatives).

We selected the Senate for two reasions:

1. The Senate provides a more represenative overview of the political sentiments present on the state scale
2. The Senate provided the lists of Roll Call votes used in our analysis in a much more accessible format (XML) than the House of Representatives (HTML)

To begin our analysis, we examined the characteristics of the senators who participated in each congress, using the Congress API provided by the NY Times. This dataset was made retreived as JSON from a URI based API. These JSON files - as well as all of the other datafiles we fetched - were cleaned by an R script, prior to storage in a SQLite database for analysis. This data was stored in the ”members” table, and contains a variety of variables, including identifiers such as name, ID, party, state, and seniority in addition to metrics like percentage of votes with party and vote miss rate.

The member data we retreived got us interested in examining the voting behaviour of the senators. We proceed with our analysis by examining the bills which were voted on in each session of the senate. We focus our analysis on roll-call bills - those bills where the vote of each senator is individually recorded. In order to examine these bills we first fetched a series of files from senate.gov containing lists of roll-call votes (and metadata) for each session or year of congress in an XML format. These data were stored in the ”senateRollCalls” table, which contains the date of each vote, the issue at hand, and the outcome - overall number of yeas and nays.

While the roll-call vote data from the senate tells us about passage rates for individual bills, it does not reveal individual voting behaviour. Indeed, this data was initially retreived in the interest of compiling a complete list of roll-call votes for lookup on our third and final data-source, govtrack.org. The govtrack group provides a bulk data API which can be used to lookup the vote of each individual senator on a given bill. This data was retreived as JSON, before being stored in our third table ”votes”. We use this information to examine each senators voting behavior individually, and in comparison to the corresponding parties or congresses.

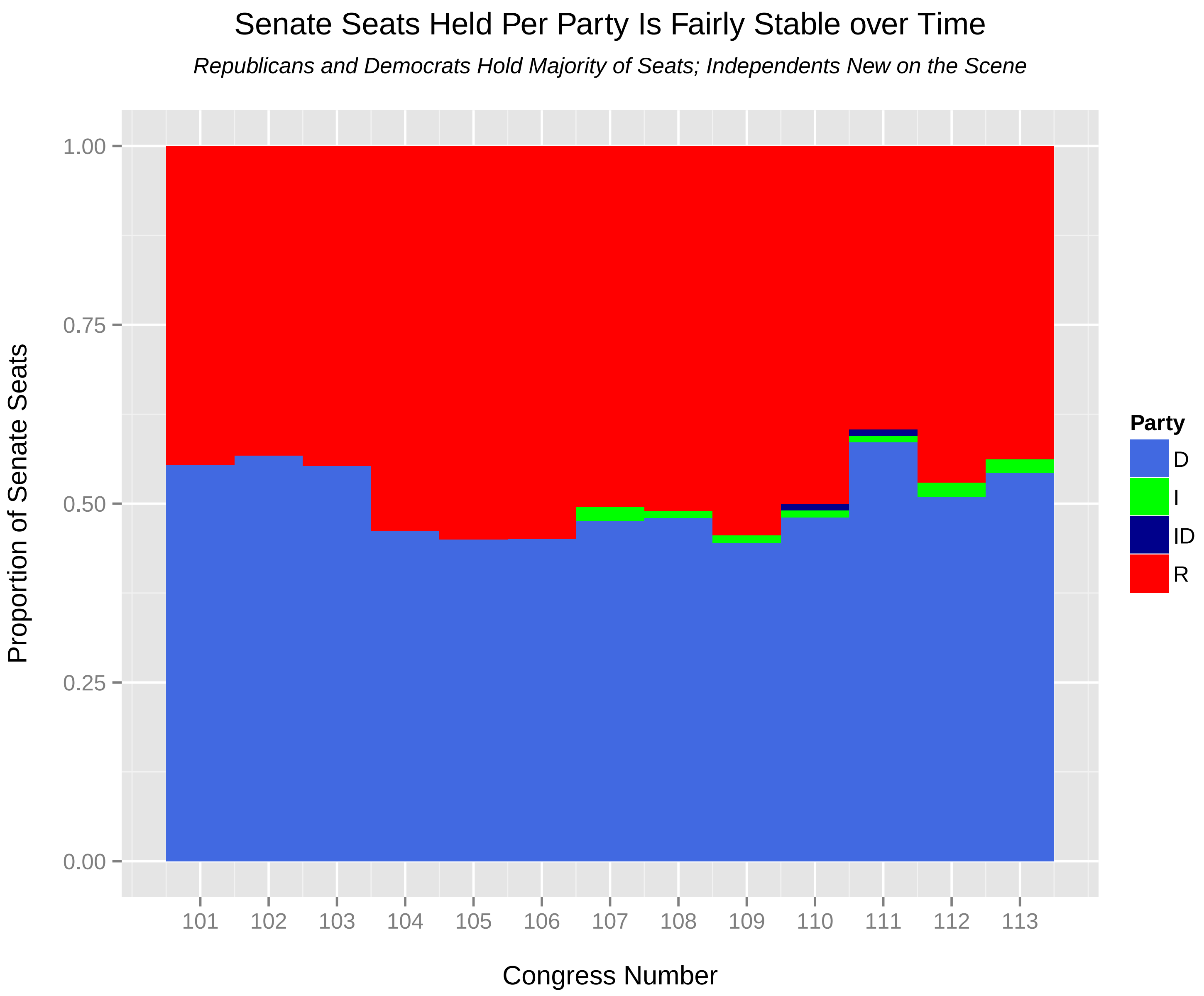
The Importance of Assessing our Government

The US government is said to be by the people, of the people, and for the people. The importance of transparency cannot be overstated in order for such a government to remain accountable to the people. It was for this reason that our group was surprised that there was no readily published workflow for citizen analysis of data on important governmental bodies like the senate. In this report we detail a protocol which can be used to fetch and store data regarding senate members, and their voting behaviours; the three-step process we detail for aquiring this data, is not very straightforward. We hope that the publication of workflows such as this will make US Government data more accessible to the public. This workflow may also serve as an entry point into more advanced analysis - such as corpus analysis and machine learning - of Senate proceedings and bills.

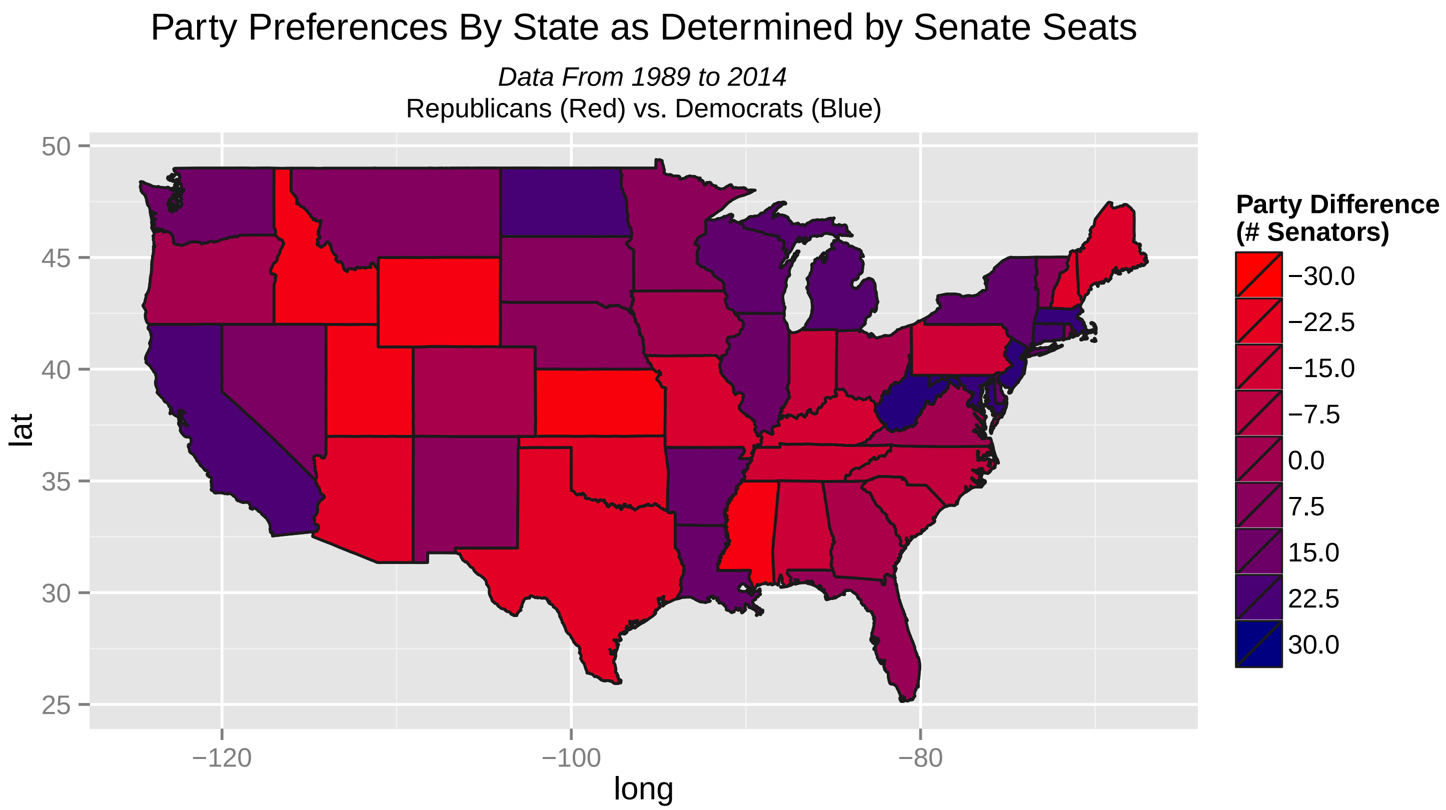
Examining The Members of Senate

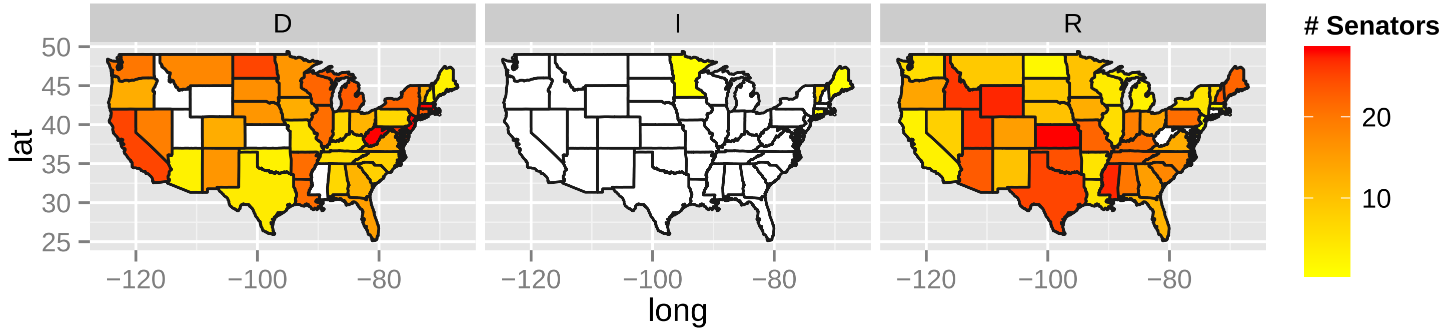
Introduction to US Political Parties

The subsequent figure is the culmination of an analysis of the power held by each of the political parties in the US Senate. We confirm the prevailing understanding of American politics as a system dominated by two major political parties - a bipartisan system. In recent years, an independent party has emerged. However, since then the party has not grown significantly; they have never held more than 2 seats. There is one minor party featured in this plot, the Iowa Democrats, who are an offshoot of the Democratic party. They have only ever held two seats in the US Senate, but do hold a good deal of power in the state; the party also has a noteworthy involvement in the Iowa Caucuses, which are important for presidential elections (http://iowademocrats.org/).

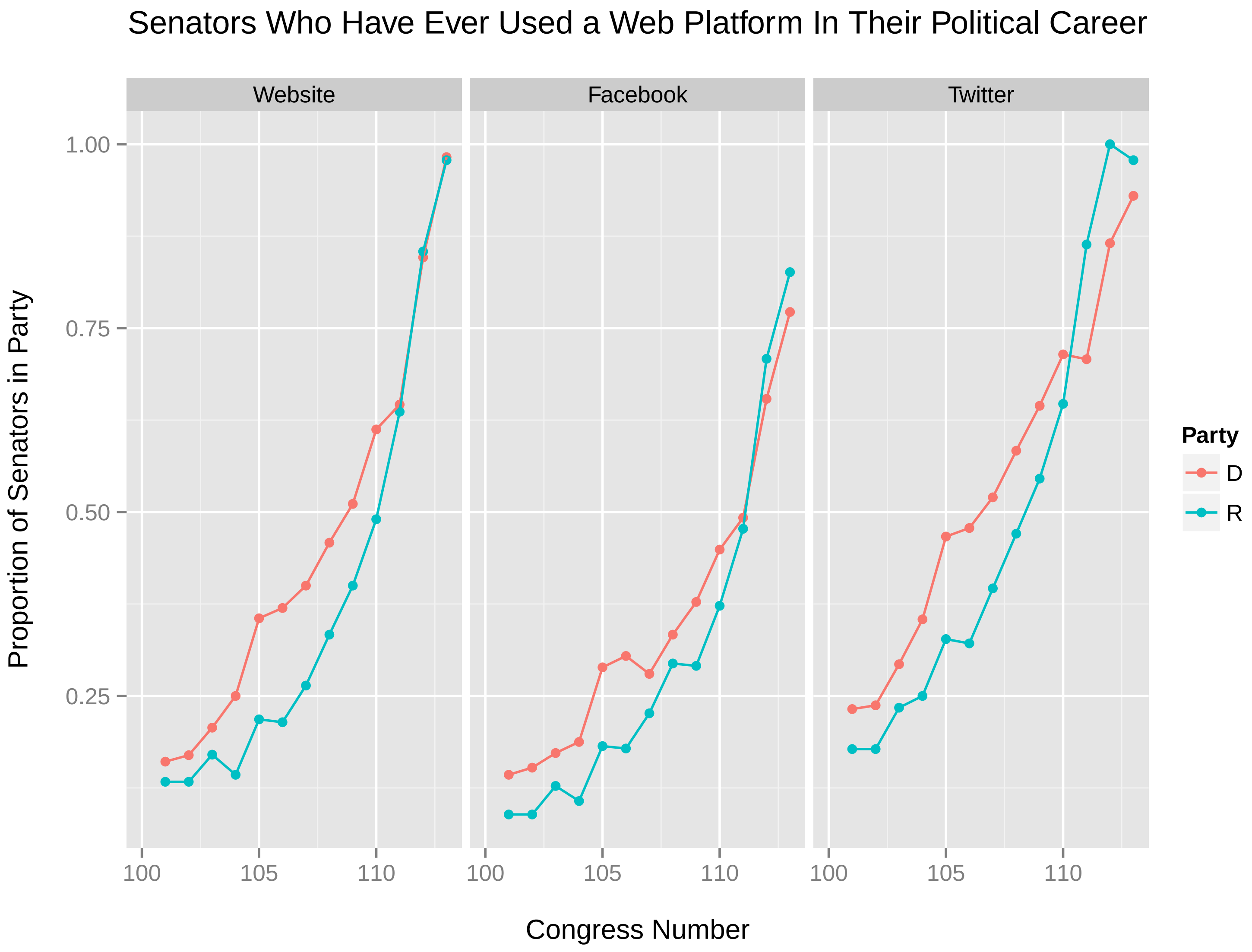


```
stateParty = queryDB("
  SELECT state as stateAbrev, party, count(*) as ct
  FROM members
  WHERE party IN (’D’, ’R’) GROUP BY party, state
")
statePartyWide = dcast(stateParty, stateAbrev~party, value.var="ct")
statePartyWide[is.na(statePartyWide)]=0
statePartyWide$diff = statePartyWide$D-statePartyWide$R
statePartyWide$state = apply(statePartyWide, 1, FUN=function(x){stateAbrevToFull(x["stateAbrev"])})
stateMap = map_data("state")
ggplot(statePartyWide)+
  geom_map(data=stateMap, map=stateMap, aes(x=long, y=lat, map_id=region), fill="#ffffff", color="grey10")+
  geom_map(data=statePartyWide, map=stateMap, aes(fill=diff, map_id=state), color="grey10")+
  scale_fill_gradient(name="Party Difference\n(# Senators)", low="red", high="navy blue",
    guide="legend", limits=c(-30, 30), breaks=seq(-30, 30, length.out = 9))+
  ggtitle(expression(atop("Party Preferences By State as Determined by Senate Seats",
    atop(italic("Data From 1989 to 2014"), "Republicans (Red) vs. Democrats (Blue)"))))
```



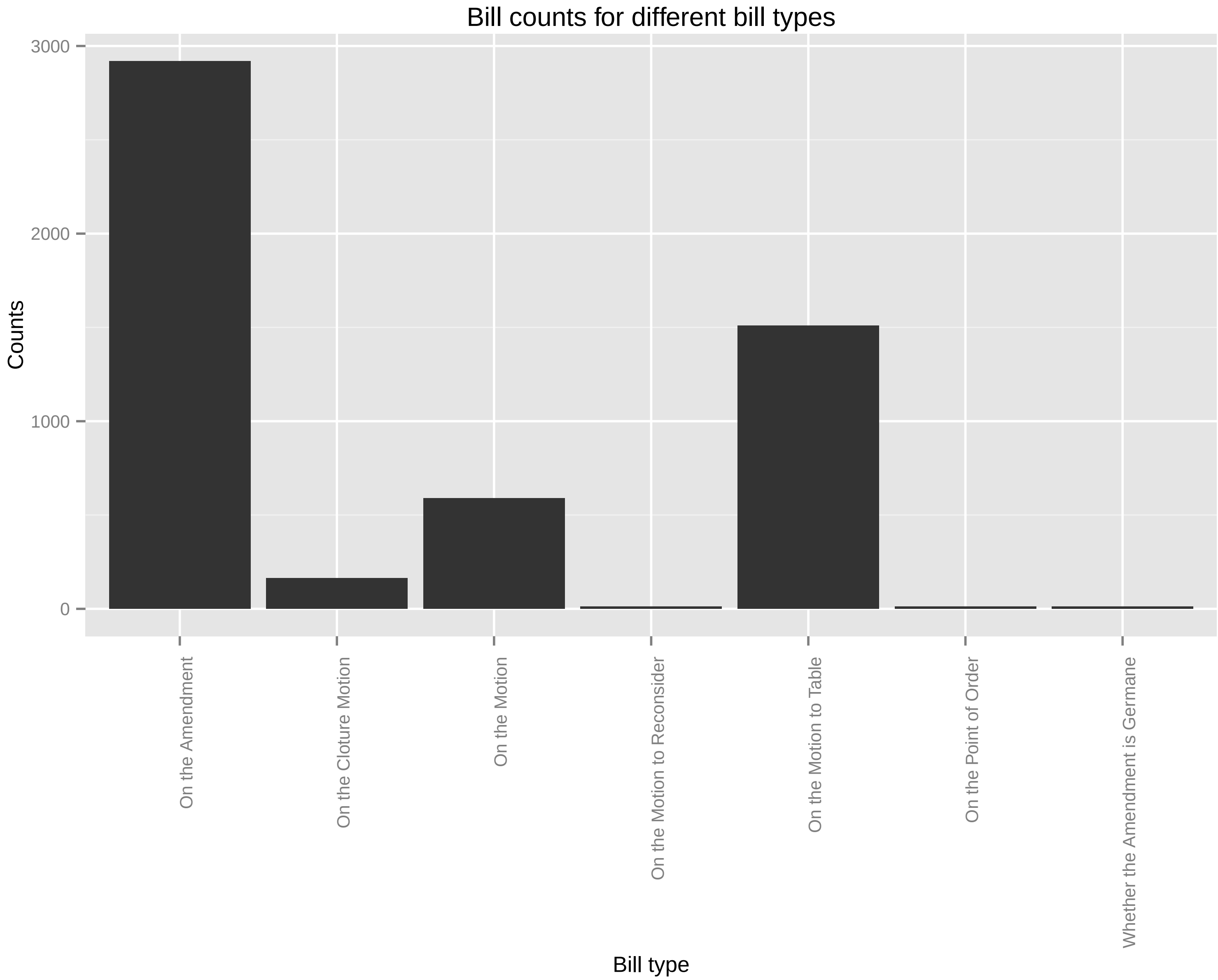


```
websiteCt = queryDB("SELECT party as Party, congressNumber, count(*) as webct FROM members WHERE URL!='' AND Par
twitterCt = queryDB("SELECT party as Party, congressNumber, count(*) as twitterct FROM members WHERE twitter_acc
fbCt = queryDB("SELECT party as Party, congressNumber, count(*) as fbct FROM members WHERE facebook_account!=''
totalCt = queryDB("SELECT party as Party, congressNumber, count(*) as totalct FROM members WHERE Party!='I' GROU
mediaCt = merge(websiteCt, twitterCt, by = c("Party", "congressNumber"))
mediaCt = merge(mediaCt, fbCt, by = c("Party", "congressNumber"))
mediaCt = merge(mediaCt, totalCt, by = c("Party", "congressNumber"))
mediaCt$Website = mediaCt$webct/mediaCt$totalct
mediaCt$Twitter = mediaCt$twitterct/mediaCt$totalct
mediaCt$Facebook = mediaCt$fbct/mediaCt$totalct
mediaPlotData = melt(mediaCt, measure.vars = c("Website", "Facebook", "Twitter"),
    id.vars = c("Party", "congressNumber"))
ggplot(mediaPlotData, aes(x = congressNumber, y = value, color = Party)) + geom_point() +
    geom_line() + facet_wrap(~variable) + xlim(100, 113) + xlab("\nCongress Number") +
    ylab("Proportion of Senators in Party\n") + ggtitle("Senators Who Have Ever Used a Web Platform In Their Pol
```

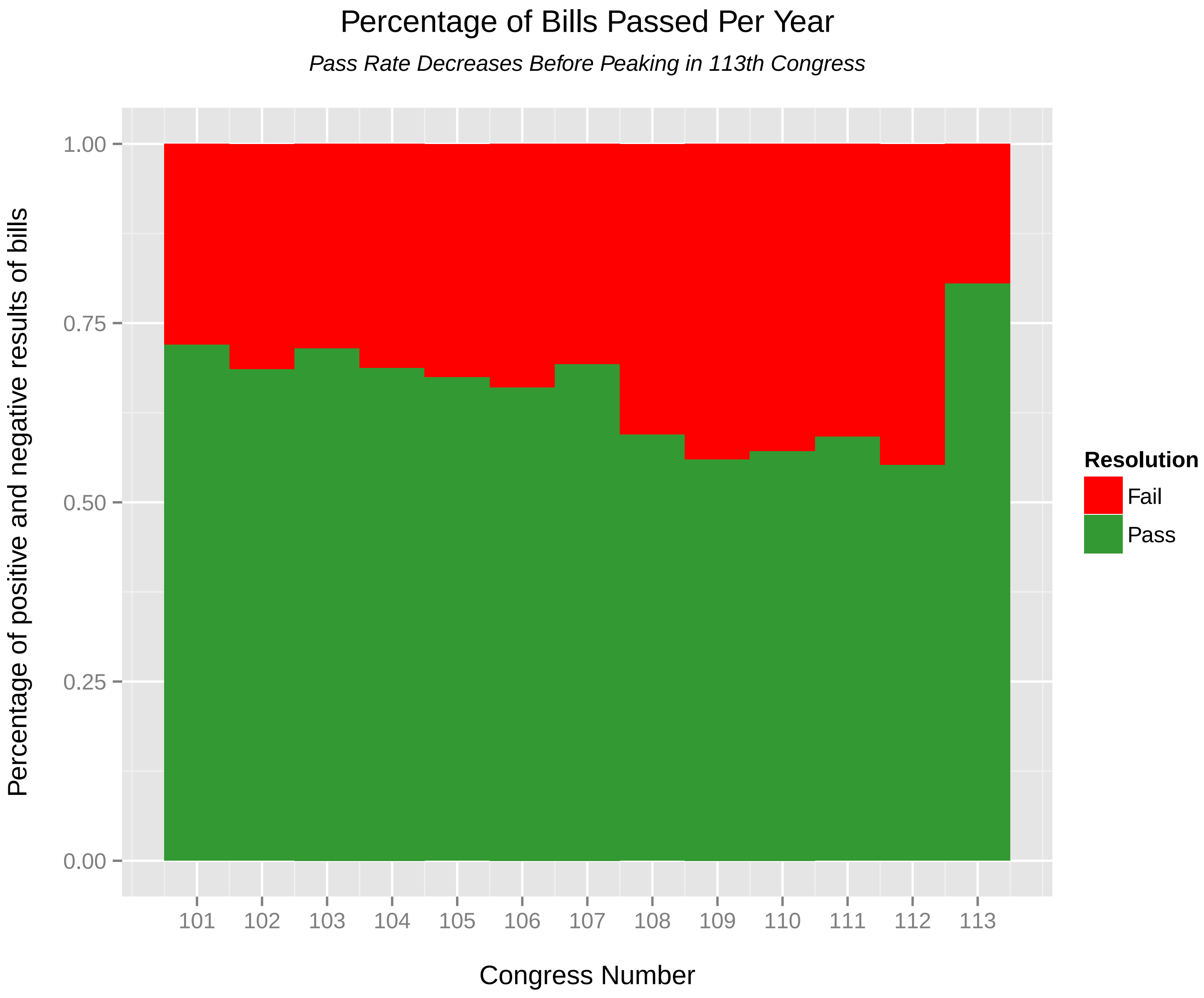


1 Examining Patterns in Bill Passage

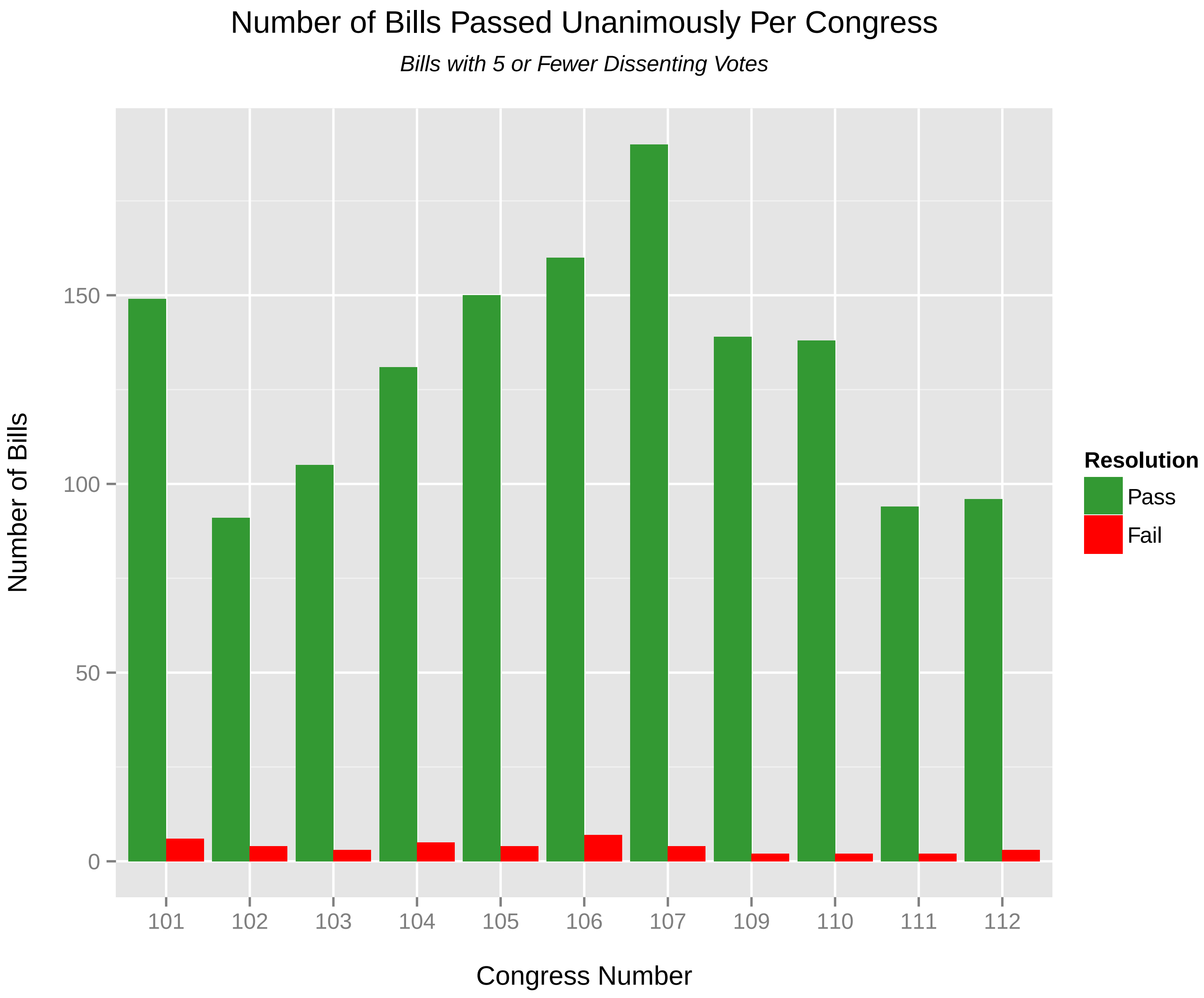
```
query <- "select * from
          (select questionType, count(*) as cnt
           from senateRollCalls
           where questionType not null
           group by questionType)
         where cnt > 10"
type <- queryDB(query, "data.sqlite")
ggplot(type) +
aes(x = questionType, y = cnt) +
labs(title="Bill counts for different bill types") +
xlab("Bill type") +
ylab("Counts") +
geom_bar(stat="identity") +
theme(text = element_text(size=10),
    axis.text.x = element_text(angle=90, hjust = 1))
```



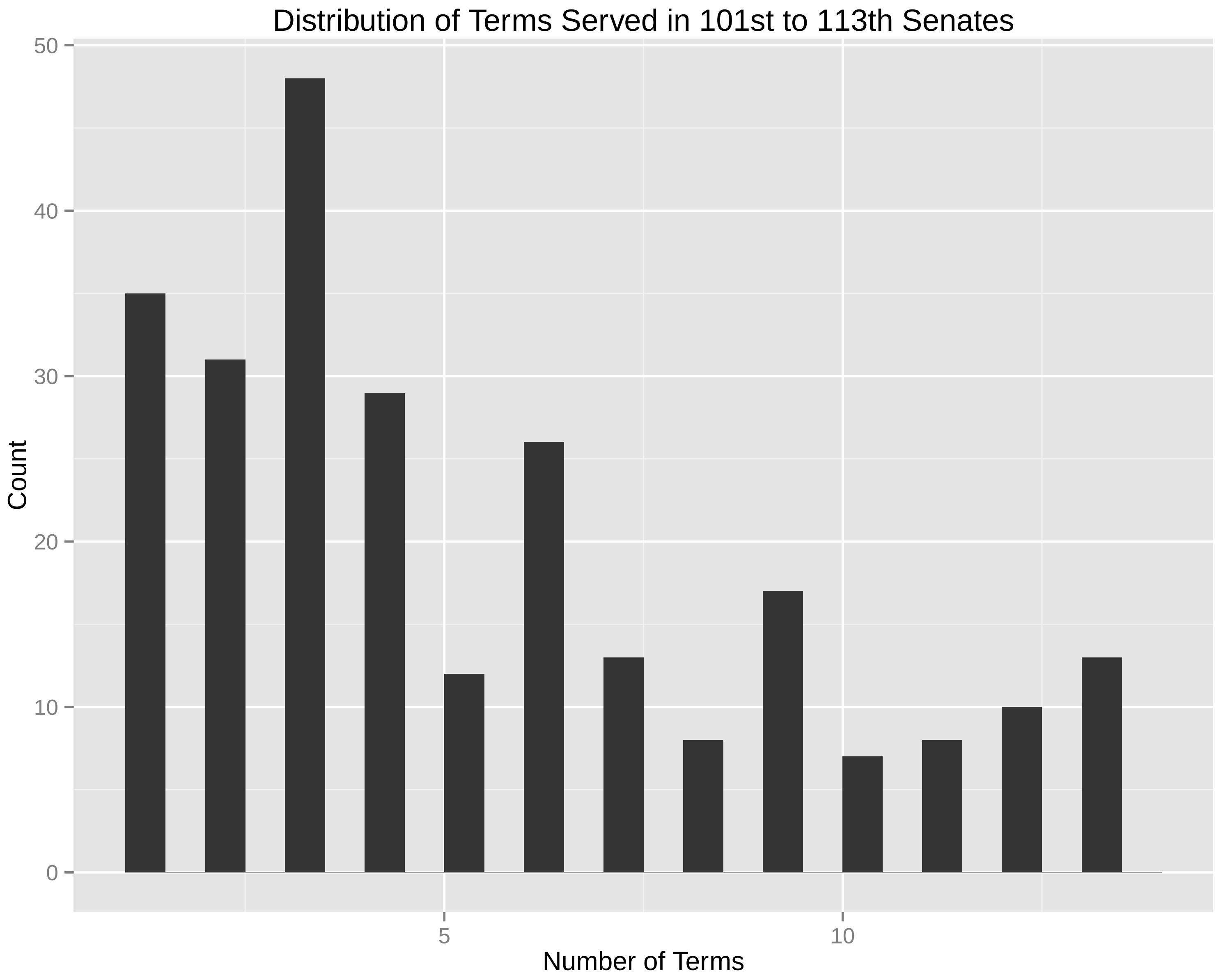
```
passedQuery <- "select 'Pass' as res, congressNumber, count(*) as cnt
          from senateRollCalls
          where result == 'Agreed to'
                or result == 'Confirmed'
                or result == 'Passed'
          group by congressNumber"
passedResults <- queryDB(passedQuery, "data.sqlite")
failedQuery <- "select 'Fail' as res, congressNumber, count(*) as cnt
          from senateRollCalls
          where result == 'Rejected'
          group by congressNumber"
failedResults <- queryDB(failedQuery, "data.sqlite")
results <- rbind(passedResults, failedResults)
ggplot(results) + aes(x=congressNumber, y=cnt, fill=res) +
  geom_histogram(position="fill", stat="identity", width=1) +
  scale_fill_manual(name = "Resolution", values = c("Pass"="#339933", "Fail"="red")) +
  xlab("\nCongress Number") +
  scale_x_continuous(breaks=101:113) +
  ylab("Percentage of positive and negative results of bills\n") +
  ggtitle(expression(atop("Percentage of Bills Passed Per Year", atop(italic("Pass Rate Decreases Before Peaki
```



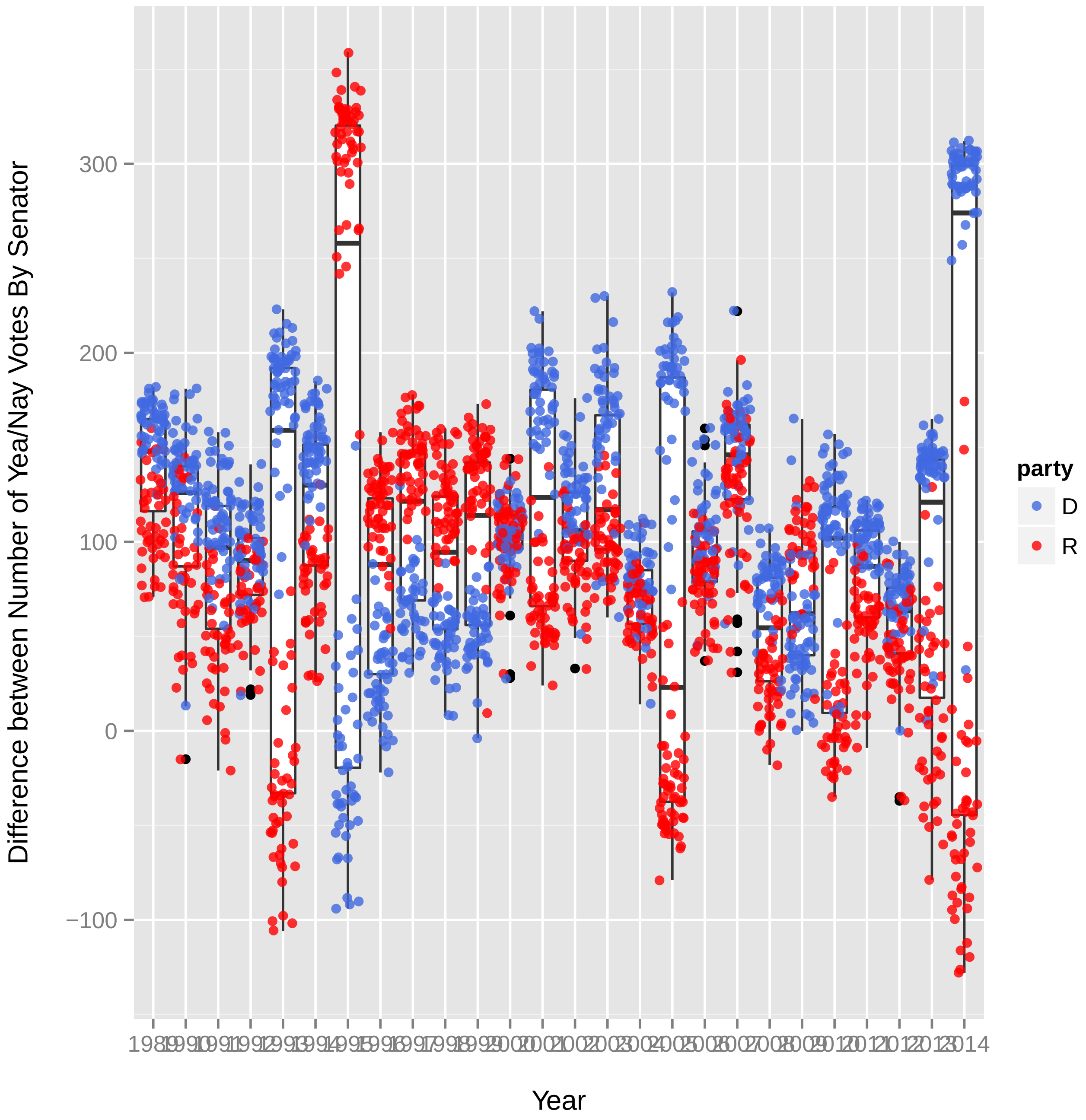
```
query <- "select pass, fail, p.congressNumber
          from
          (select congressNumber, count(*) as Pass
           from senateRollCalls
           where  CAST(nays as INTEGER) <= 5
           group by congressNumber) as p
          join
          (select congressNumber, count(*) as Fail
           from senateRollCalls
           where  CAST(yeas as INTEGER) <= 5
           group by congressNumber) as f
          on p.congressNumber == f.congressNumber"
unanimous <- queryDB(query, "data.sqlite")
unanimous$congressNumber <- factor(unanimous$congressNumber)
unanimous <- melt(unanimous[,c('congressNumber','Pass','Fail')],id.vars = 1)
ggplot(unanimous,aes(x = congressNumber,y = value)) +
  geom_bar(stat='identity',aes(fill = variable),position = "dodge")+
  scale_fill_manual(name = "Resolution", values = c("Pass"="#339933", "Fail"="red")) +
  xlab("\nCongress Number") +
  ylab("Number of Bills\n") +
  ggtitle(expression(atop("Number of Bills Passed Unanimously Per Congress", atop(italic("Bills with 5 or Fewe
```



```
query = "SELECT id, first_name, last_name, party, seniority, count(*) AS ct FROM members GROUP BY id ORDER BY
senatorTotals = queryDB(query, "data.sqlite")
ggplot(senatorTotals, aes(x=ct))+xlim(1, 14)+geom_bar(binwidth=.5)+xlab("Number of Terms")+ylab("Count")+ggtit
```

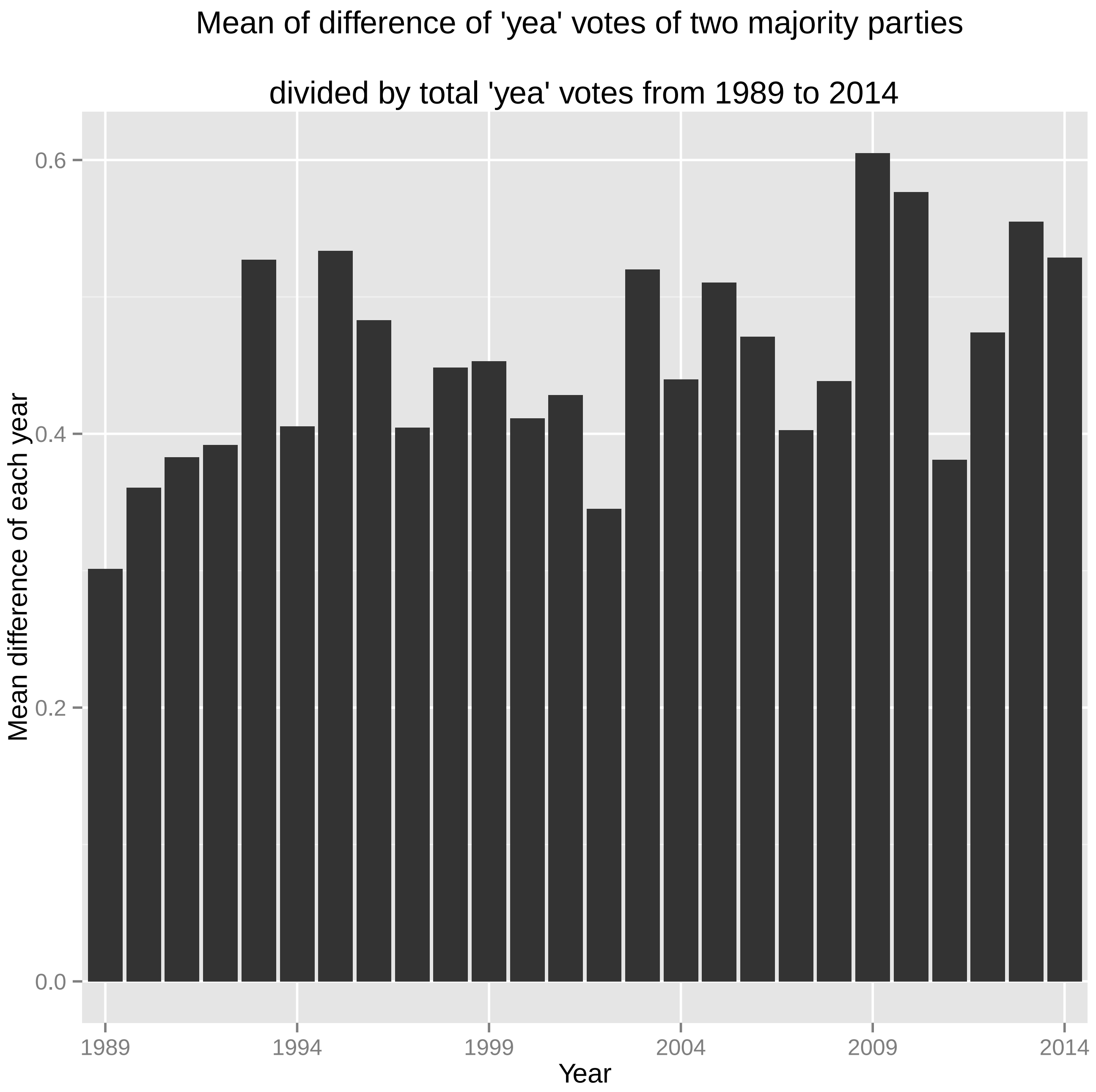


```
  votedata = queryDB("SELECT id, party, vote, count(*) as ct, year FROM votes WHERE party in (’D’, ’R’) GROUP BY
  votedatawide = dcast(votedata, id+year+party~vote, value.var="ct")
  votedatawide$diff = (votedatawide$Yea-votedatawide$Nay)
  ggplot(votedatawide, aes(x=year, y=diff))+geom_boxplot()+geom_jitter(aes(color=party), alpha=.8)+scale_color_m

## Warning:  Removed 3 rows containing non-finite values (stat_boxplot).
## Warning:  Removed 3 rows containing missing values (geom_point).
```

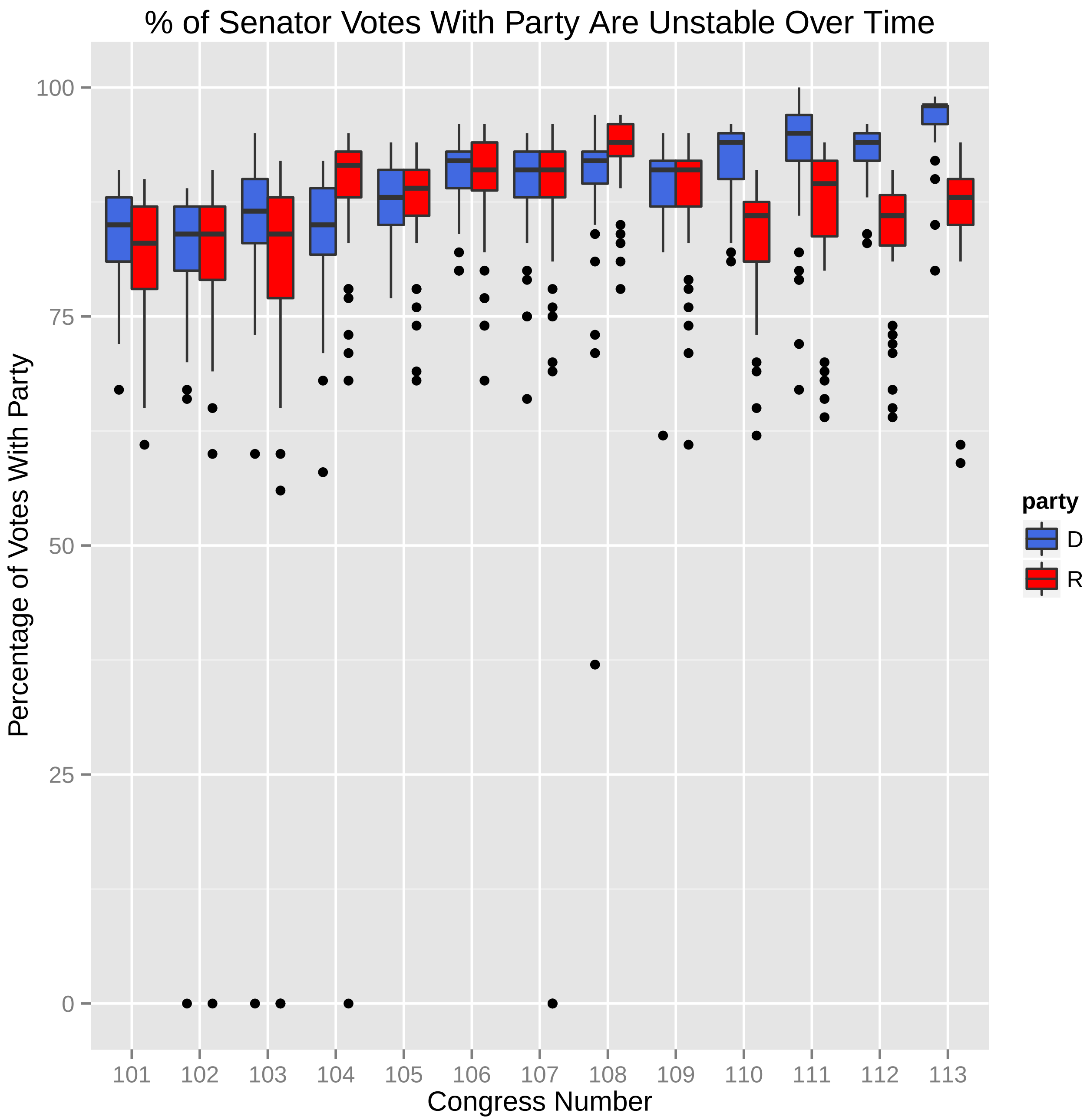


```
query = "SELECT party, yeas, nays, (yeas+nays) as total, (100*(yeas-nays)/(yeas+nays)) as voteDiff, congressNumb
rollCallStats = queryDB(query, "data.sqlite")

## Error in sqliteSendQuery(conn, statement): error in statement: no such column: party

rollCallStats$year = apply(rollCallStats, 1, function(x) {
    congressToYear(x["congressNumber"], x["sessionNumber"])
})

## Error in apply(rollCallStats, 1, function(x) {: object 'rollCallStats' not found

ggplot(rollCallStats, aes(x = as.factor(year), y = voteDiff)) + geom_boxplot() +
    geom_smooth(aes(group = 1), method = "lm") + geom_point(alpha = 0.1) + ggtitle("Roll-call Vote Disagreement
    xlab("\nYear") + ylab("Percentage Difference\n100*(Yea-Nay)/Total")

## Error in ggplot(rollCallStats, aes(x = as.factor(year), y = voteDiff)): object 'rollCallStats' not found
```

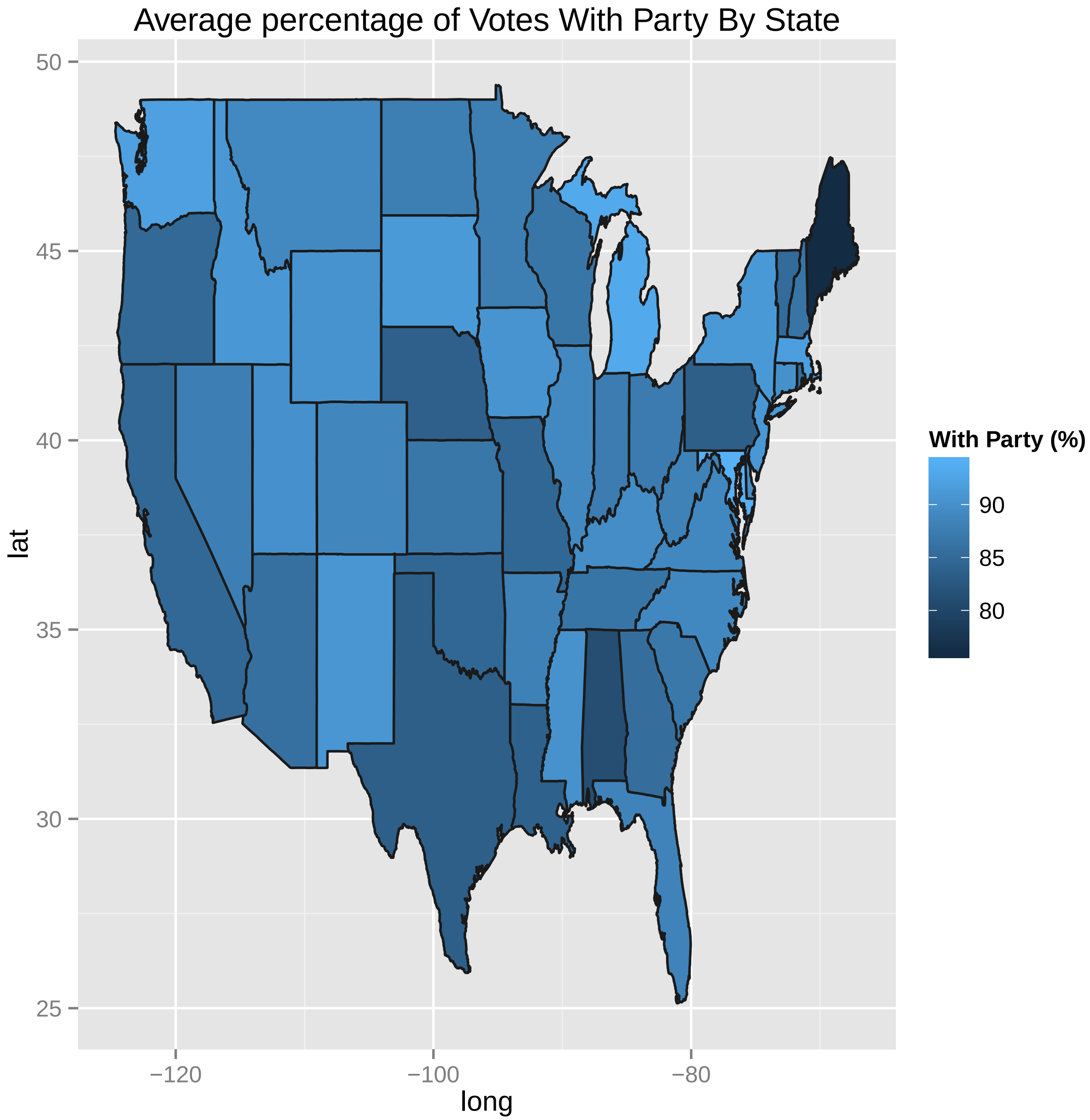
```
query = "select r.year as year, r.voteNumber as voteNumber,
            abs(r.c - d.c) * 1.0 / (r.c + d.c) as diff
         from (select voteNumber, year, count(*) as c
               from votes
               where vote == ’Yea’ and party == ’R’ group by year, voteNumber)
               as r
              join
            (select voteNumber, year, count(*) as c
               from votes
               where vote == ’Yea’ and party == ’D’ group by year, voteNumber)
               as d
            on r.voteNumber == d.voteNumber and r.year == d.year"
yeaDiff = queryDB(query, "data.sqlite")
yeaDiffMean = setNames(aggregate(diff ~ year, yeaDiff, mean), c("year", "mean"))
yeaDiffSd = setNames(aggregate(diff ~ year, yeaDiff, sd), c("year", "std"))
yeaDiffDistribution = merge(yeaDiffMean, yeaDiffSd, by="year")
ggplot(yeaDiffDistribution) +
aes(x = year, y = mean) +
scale_x_discrete(breaks=c(1989,1994,1999,2004,2009,2014)) +
labs(title="Mean of difference of ’yea’ votes of two majority parties
     \ndivided by total ’yea’ votes from 1989 to 2014") +
xlab("Year") +
ylab("Mean difference of each year") +
geom_bar(stat="identity")
```


Mean of difference of 'yea' votes of two majority parties
divided by total 'yea' votes from 1989 to 2014
Mean difference of each year
0.6
0.4
0.2
0.0
1989
1994
1999
2004
2009
2014
Year

```
memberPct = queryDB("SELECT id, party, missed_votes_pct as missed, votes_with_party_pct as withParty, next_ele
memberPct$withParty = as.integer(as.character(memberPct$withParty))
memberPct$congressNumber = as.factor(memberPct$congressNumber)
ggplot(memberPct, aes(x=congressNumber, y=withParty, fill=party))+geom_boxplot()+xlab("Congress Number")+ylab(
```



```
stateMap = map_data("state")
stateVote = queryDB(sprintf("SELECT state as stateAbrev, avg(votes_with_party_pct) as withParty FROM members (
stateVote$state = apply(stateVote, 1, FUN=function(x){stateAbrevToFull(x["stateAbrev"])})
ggplot()+geom_map(data=stateMap, map=stateMap, aes(x=long, y=lat, map_id=region), fill="#ffffff", color="grey1
```


Average percentage of Votes With Party By State
50
45
40
35
30
25
lat
−120
−100
−80
long
With Party (%)
90
85
80