Problem Set 1

Data Visualisation for Social Scientists

Jeffrey Ziegler

Instructions

- *Please show your work! You may lose points by simply writing in the answer. If the problem requires you to execute commands in `R`, please include the code you used to get your answers. Please also include the `.R` file that contains your code. If you are not sure if work needs to be shown for a particular problem, please ask.*
- *Your homework should be submitted electronically on GitHub.*
- *This problem set is due before 23:59 on Wednesday January 28, 2026. No late assignments will be accepted.*

Roll Call Votes in the European Parliament

Data Manipulation

First, you need to download data from the first European Parliament, including information on each MEP from EP1 and how they voted in each recorded roll-call vote during EP1.

1. *Load these datasets into your global environment:*

   - ***`mep_info_26Jul11.xls`*** *(includes MEP characteristics from EP1–EP5, you need "Sheet = EP1")*
   - ***`rcv_ep1.txt`*** *(EP1 roll-call votes)*

```
# load MEP info (EP1 sheet only)
# I'm assuming all datasets have been downloaded into Downloads folder
mep_info <- read_excel("~/Downloads/mep_info_26Jul11.xls", sheet = "EP1")

# load roll-call votes (comma-delimited text)
rcv_ep1 <- read_delim("~/Downloads/rcv_ep1.txt", delim = ",", col_names =
    TRUE)
```

2. *Briefly describe (2–3 sentences each) the unit of analysis and key variables in each of these two datasets.*

   - `mep_info` (EP1 sheet): The unit of observation is an individual MEP, while the main variables of interest are MEPID (unique ID), MEPNAME, MS (member state code),, NP (national party code), EPG(EP party group code), nomdim1 (NOM-D1), and nomdim2 (NOM-D2).
   - `rcv_ep1`: The unit of analysis is MEP-vote pair in wide format (row per MEP, columns $V_1 - V_n$ for votes). Key variables: MEPID, MEPNAME, MS, NP, EPG (metadata), $V_1 - V_n$ (vote codes: e.g., Y=Yes, N=No, Abs=Abstain).

3. *The **`rcv_ep1`** data are in a wide format, with V1, V2, . . . , Vn as separate vote columns.*

   - *Identify which columns are ID/metadata (MEPID, MEPNAME, MS, NP, EPG) and which columns are vote decisions ($V_1 \ldots V_n$). Tidy the voting data such that each row/observation is a single vote for a single MEP.*

```
# identify vote columns (starting from V1)
vote_cols <- str_subset(names(rcv_ep1), "^V\\d+$")

# pivot to long format
rcv_long <- rcv_ep1 %>%
  pivot_longer(cols = all_of(vote_cols),
               names_to = "vote_id",
               values_to = "decision") %>%
  mutate(vote_num = as.integer(str_remove(vote_id, "V")))
```

   - *Create a summary table of counts of decision categories (e.g. Yes/No/Abstain/Present but did not vote/Absent) across all votes.*

```
# create summary table of decision counts
xtable(rcv_long %>%
         count(decision))
```

Table 1: Raw counts of vote decision for each MEP.

Decision	N
0	99753
1	88185
2	75171
3	9577
4	109224
5	103618

4. *Construct a new dataset that combines MEP-level information with their vote decisions from EP1 in long format (from part 3). Check for missingness.*

```r
# need to rename first column of mep_info for join
names(mep_info)[1] <- "MEPID"
# drop any invalid votes if present
votes_joined <- rcv_long %>%
  left_join(mep_info %>% select(MEPID, `NOM-D1`, `NOM-D2`), by = "MEPID")

# check for missingness
# convert NOM-1 and NOM-2 to numeric
votes_joined$`NOM-D1` <- as.numeric(votes_joined$`NOM-D1`)
votes_joined$`NOM-D2` <- as.numeric(votes_joined$`NOM-D2`)

# now look at the rows for which we have NAs in "NOM-D1" & "NOM-D2"
missing_check <- votes_joined %>% filter_at(vars(`NOM-D1`, `NOM-D2`), any_vars(is.na(.)))

# look at which MEPs this is for
xtable(unique(missing_check[, c("MEPNAME", "EPG")]) %>%
         count(EPG))
```

It turns out that we do have missingness along the two Nominate dimensions for 48 MEPs, the majority of whom represent the G and S European Party Groups.

EPG	n
0	1
E	7
G	10
L	2
M	7
N	4
R	1
S	16

5. *Compute, for each EP group in EP1:*

   - *The mean rate of Yes votes (Yes over Yes+No+Abstain) across all roll calls.*
   - *The mean abstention rate.*
   - *The mean vote preferences along the two contested dimensions (NOM-D1 and NOM-D2).*

```r
# create yes or "other" variable first
votes_joined$yes_decision <- ifelse(votes_joined$decision==1, 1, 0)
# then the abstention variable
votes_joined$abstain_decision <- ifelse(votes_joined$decision==3, 1, 0)

# get the mean:
# yes across all roll-calls
# abstention across all roll-calls
# NOM-1 and NOM-2
aggregate(votes_joined[,c("yes_decision", "abstain_decision", "NOM-D1", "NOM-D2")], by=list(votes_joined$EPG), FUN=mean, na.rm=TRUE)
```

Table 2: Mean "Yes" rate, abstention rate, and vote preference by EP group.

EP Group	Yes rate	Abstain rate	NOM-D1	NOM-D2
C	0.25835752	0.046794941	0.8109206	0.53011111
E	0.21822883	0.009206226	0.5122441	-0.27672441
G	0.08430661	0.011482972	0.2796944	-0.81805556
L	0.14718313	0.019139330	0.4085111	-0.32422222
M	0.14323438	0.021700243	-0.3567391	-0.20130435
N	0.09015801	0.008713318	0.2501905	-0.38561905
R	0.08039590	0.046622678	-0.5864167	-0.04191667
S	0.19473605	0.019432130	-0.0979600	0.26092000

Data Visualization

1. *Plot the distribution of the first NOMINATE dimension by EP group, and explain any trends you see.*

```r
# again, there's going to be some non-numeric values that go to NA
mep_info$`NOM-D1` <- as.numeric(mep_info$`NOM-D1`)
mep_info$`NOM-D2` <- as.numeric(mep_info$`NOM-D2`)
```

```r
ggplot(mep_info, aes(x=`NOM-D1`, y=`EP Group`, fill=`EP Group`)) +
  geom_density_ridges(alpha = 0.5) +
  labs(x="NOM-D1", y="") +
  scale_fill_brewer(palette="Set1") + theme_minimal()
```

We can see in Figure 1 that EP groups show ideological clustering on NOM-D1 (left-right) that we might expect such as Social Democrats (Socialist Group, Party of European Socialists) on the left (Party of European Socialists), Conservatives on the right (Progressive European Democrats, European People's Party). Most of the parties are quite cohesive (i.e., normally distributed around a central ideological point), though a couple of parties are more unequally distributed ideologically.

2. *Make a scatterplot of nomdim1 (x-axis) and nomdim2 (y-axis), with one point per MEP and color by EP group.*

```r
ggplot(mep_info, aes(x=`NOM-D1`, y=`NOM-D2`, color=`EP Group`)) +
  geom_point(alpha=0.6) +
  labs(x="NOM-D1", y="NOM-D2") + lims(x=c(-1,1), y=c(-1,1)) +
  scale_color_brewer(palette="Set1") + theme_minimal()
```

Figure 1: NOM-D1 Distribution by EP Group

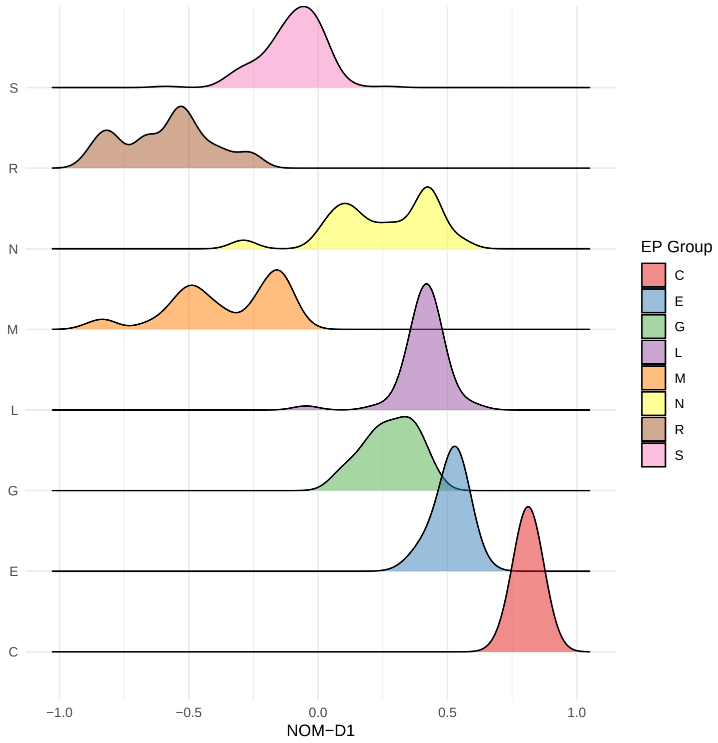
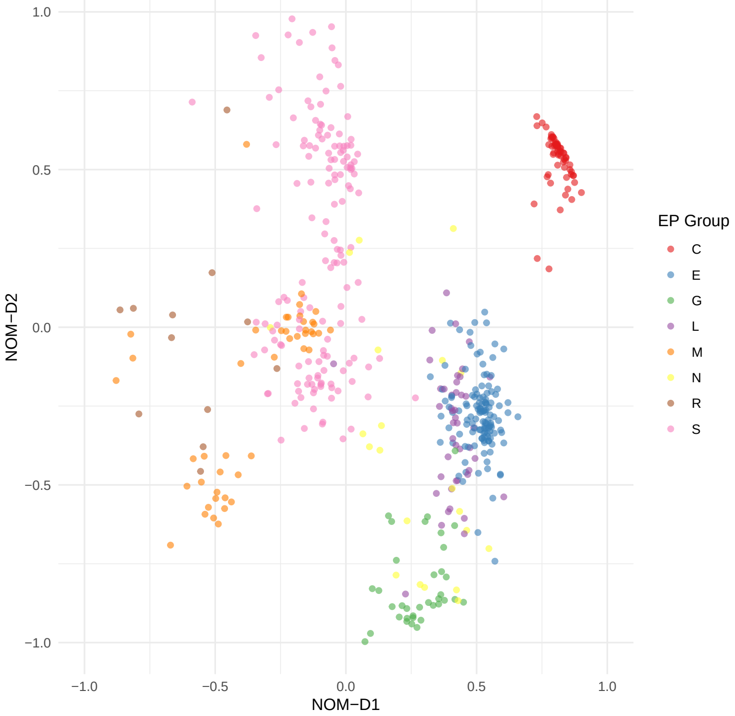


Figure 2: Individual MEPs plotted along NOM-D1 and NOM-D2 distinguished by EP Group.



3. *Produce a boxplot of the proportion voting Yes for each vote by EP group to visualize cohesion.*

```r
# get proportion that voted yes for each vote by EP group
prop_votes_EPG <- votes_joined %>%
  group_by(vote_id, yes_decision, EPG) %>%
  summarise(N=n()) %>%
  group_by(vote_id, EPG) %>%
  mutate(prop_vote=N/sum(N))

# this gives us a dataframe with proporitions of each vote type, yes and no
prop_yes_EPG <- prop_votes_EPG[which(prop_votes_EPG$yes_decision==1),]
# we want to isolate the proportion of "yes" votes
# so what do we do if no one in an EPG voted yes?
# we'll start by creating an empty data frame with an observation for
# all parties for all votes filled with zeroes
no_votes <- expand.grid(vote_id = unique(prop_votes_EPG$vote_id),
                        EPG = unique(prop_votes_EPG$EPG),
                        prop_vote_no = 0)
# then we'll merge that with out previous dataframe
prop_yes_EPG <- merge(prop_yes_EPG[,c("vote_id", "EPG", "prop_vote")], no_votes, by=c("vote_id", "EPG"), all.y = T)
# so now we'll have an observation for EPGs that had no one that vote yes (prop_vote==1)
prop_yes_EPG$prop_vote <- ifelse(is.na(prop_yes_EPG$prop_vote), 0, prop_yes_EPG$prop_vote)
```

```r
# plot only the yes proportions
ggplot(prop_yes_EPG[,c("vote_id", "EPG", "prop_vote")],
       aes(x = EPG, y = prop_vote, fill = EPG)) +
  geom_boxplot() +
  labs(y = "Proportion Yes", x="") +
  scale_color_brewer(palette="Set1") + theme_minimal()
```

4. *Display the proportion voting Yes across votes by national party using a bar plot.*

```r
# do similar thing as previous problem, but now with NP
# we don't have the same issue where we want to get the proportion for each vote
prop_yes_NP <- votes_joined %>%
  group_by(yes_decision, NP) %>%
  summarise(N=n()) %>%
  group_by(NP) %>%
  mutate(prop_vote=N/sum(N))

# just need to make our NP variable a factor
prop_yes_NP$NP <- as.factor(prop_yes_NP$NP)
```

```r
ggplot(prop_yes_NP[which(prop_yes_NP$yes_decision==1),],
       aes(x = NP, y = prop_vote, fill = NP)) +
  geom_col() +
  labs(y = "Proportion Yes", x="") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Figure 3: Proportion of MEPs that voted yes distinguished by EP Group.

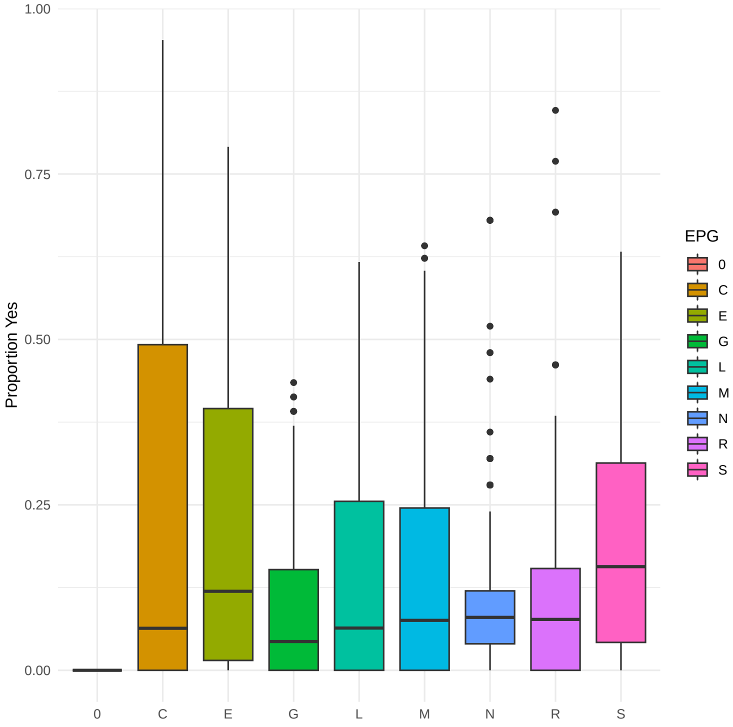
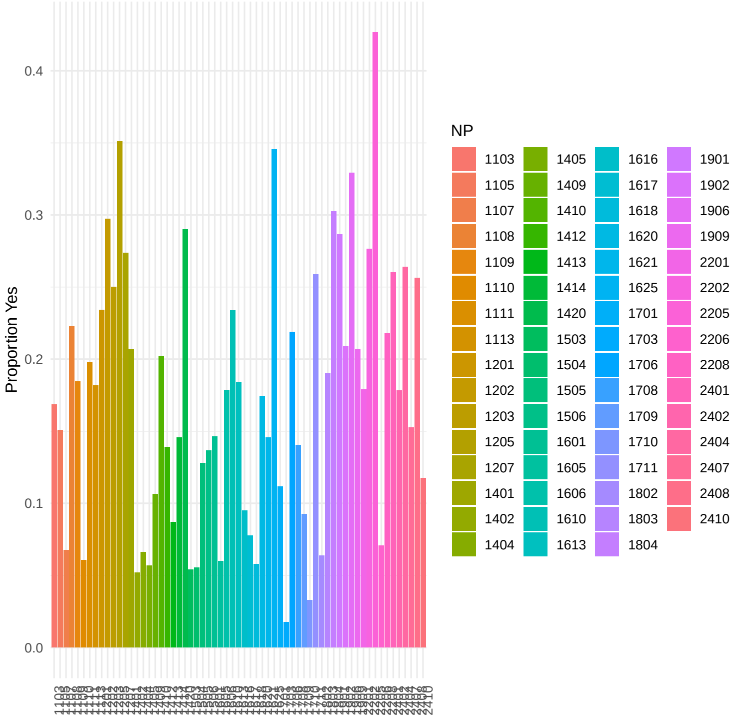


Figure 4: Proportion of MEPs that voted yes distinguished by EP Group.


Proportion Yes
NP