```r
1  # NZZ Storytelling, script for the following article:
   https://www.nzz.ch/international/freund-und-feind-betrachtet-durch-die-brille-des-
   weissen-hauses-ld.1349175#subtitle-die-methodik-im-detail
2  # questions and comments: marie-jose.kolly@nzz.ch
3
4  ### MISE EN PLACE ###
5  library(dplyr)
6  library(tibble)
7  library(sotu)
8  library(tm)
9  library(ggplot2)
10 library(countrycode)
11 library(tidyr)
12 library(directlabels)
13
14 setwd("mypath/graphics")
15
16
17 #---------------------------------------------------------------------------
   -----------------------------
18 ### VIEW AND CONFIGURE DATA ###
19 nrow(sotu_meta)
20 length(sotu_text)
21 head(sotu_text)
22
23 # write all sotu-addresses to directory
24 sotu_dir("mypath/data/sotu_originals", filenames)
25
26
27 #---------------------------------------------------------------------------
   -----------------------------------------------------
28 ### CORPUS CREATION AND PROCESSING ###
29
30 # Read in as corpus
31 directoryIn<-"mypath/data/sotu_red_augm"
32 docs<-Corpus(DirSource(directoryIn, encoding = "UTF-8"), readerControl = list(language
   = "eng"))
33
34
35 ## Create different corpora containing different types of information
36
37 # Corpus 1: keep punctuation but remove whitespace
38 corpus1<-tm_map(docs, stripWhitespace)
39 directoryOut<-"mypath/data/sotu_corpus1"
40 writeCorpus(corpus1, path=directoryOut, filenames=paste0(substr(dir(directoryIn), 1,
   nchar(dir(directoryIn))-4),".txt",sep=""))
41
42 # Corpus 2: remove punctuation and whitespace and convert to lowercase to count words,
   and for concept and frequent word analysis
43 corpus2<-tm_map(docs, removePunctuation, preserve_intra_word_dashes = TRUE)
44 corpus2<-tm_map(corpus2, stripWhitespace)
45 corpus2<-tm_map(corpus2, content_transformer(tolower))
46 directoryOut<-"mypath/data/sotu_corpus2"
47 writeCorpus(corpus2, path=directoryOut, filenames=paste0(substr(dir(directoryIn), 1,
   nchar(dir(directoryIn))-4),".txt",sep=""))
48
```

```
49   # Corpus 3: additionally stemmed and stopped
50   corpus3<-tm_map(corpus2, removeWords, stopwords("english"))
51   corpus3<-tm_map(corpus3, stemDocument, language = "english")
52   directoryOut<-"mypath/data/sotu_corpus3"
53   writeCorpus(corpus3, path=directoryOut, filenames=paste0(substr(dir(directoryIn), 1,
     nchar(dir(directoryIn))-4),".txt",sep=""))
54
55
56   ## Create metadata frame
57
58   # to count words and characters, read in these files again with scan, which creates a
     vector with one-word-one-element
59   txtCleanCorpus <- list.files(path="mypath/data/sotu_corpus2/", pattern="*.txt",
     full.names=F, recursive=FALSE)
60   directoryCorpus2<-"mypath/data/sotu_corpus2/"
61
62   # build metadata from filenames and sotu_meta
63   filename<-unlist(lapply(strsplit(as.character(txtCleanCorpus), "\\."),'[[', 1))
64   year<-substr(filename, nchar(filename)-3, nchar(filename))
65   president<-substr(filename, 1, nchar(filename)-5) %>% gsub("-", " ", .) %>%
     gsub("\\s$", "", .)
66
67   party<-c()
68   for (i in 1:length(president)) party[i]<-sotu_meta$party[grep(president[i],
     tolower(gsub("\\.", "", sotu_meta$president)))[1]]
69
70   metadata<-data.frame(president, year, party)
71
72   # add party information for trump as he's not in sotu_meta
73   metadata[grep("trump", metadata$president),3]<-"Republican"
74
75
76   #----------------------------------------------------------------------------------
     ---------------------------------------------------------
77   ### ANALYSIS NUMBER OF WORDS ###
78
79   # number of characters
80   nChars<-c()
81   for (i in 1:length(txtCleanCorpus)){
82     nCharsi<-sum(nchar(scan(paste0(directoryCorpus2,"/",txtCleanCorpus[i],sep=""),
     quote=NULL, what="x")))
83     nChars<-rbind(nChars, nCharsi)
84   }
85
86   # number of words
87   nWords<-c()
88   for (i in 1:length(txtCleanCorpus)){
89     nWordsi<-length(scan(paste0(directoryCorpus2,"/",txtCleanCorpus[i],sep=""),
     quote=NULL, what="x"))
90     nWords<-rbind(nWords, nWordsi)
91   }
92
93   metadata<-mutate(metadata, nWords=as.numeric(as.character(nWords)),
     nChars=as.numeric(as.character(nChars)))
94
95   #total number of words
96   sum(metadata$nWords)
```

```r
 97
 98  # plot
 99  ggplot(metadata, aes(as.numeric(as.character(year)),nWords)) +
100    geom_bar(stat="identity") +
101    ggtitle("Anzahl Wörter in den Reden der US-Präsidenten")+
102    ylab("Absolute Anzahl Wörter")+
103    theme_minimal() +
104    theme(axis.text=element_text(family="GT America", color="#05032d", size=11),
105          axis.title.x=element_blank(),
106          axis.title.y=element_text(family="GT America", color="#05032d", size=13),
107          plot.title = element_text(family="GT America", color="#05032d", size=20), # or
     play around with something like hjust=-0.12*(nchar(indicator)/10)
108          plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1),
109          plot.caption = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1, vjust=-3),
110          panel.grid = element_line(color="#ececf0", size=.3),
111          plot.margin=unit(c(1,1,1.5,1.2),"cm"))
112
113
114  #-----------------------------------------------------------------------------
     -------------------------------------------------------------
115  ### ANALYSIS COUNTRIES AND REGIONS ###
116
117
118  ## AMERICA
119  americPerFile<-read.table("mypath/data/datafiles/americUSPerFile.txt", sep="\t")
120  colnames(americPerFile)<-c("file", "americ", "ourRepublic", "our_federalUnion", "US")
121  metadata_america<-metadata %>%
122    as.tibble(.) %>%
123    mutate(americ=100*(americPerFile$americ/nWords)) %>%
124    mutate(ourRepublic=100*(americPerFile$ourRepublic/nWords)) %>%
125    mutate(our_federalUnion=100*(americPerFile$our_federalUnion/nWords)) %>%
126    mutate(US=100*(americPerFile$US/nWords)) %>%
127    gather(key=word, value=prop, americ, ourRepublic, our_federalUnion, US)
128
129  # plot
130  ggplot(metadata_america, aes(as.numeric(as.character(year)),prop, fill=word)) +
131    geom_bar(stat="identity") +
132    ggtitle("America(n), United States, Union und Republic in den Reden der US-
     Präsidenten")+
133    ylab("Proportional zur Anzahl Wörter in Rede, in Prozent")+
134    theme_minimal() +
135    theme(axis.text=element_text(family="GT America", color="#05032d", size=11),
136          axis.title.x=element_blank(),
137          axis.title.y=element_text(family="GT America", color="#05032d", size=13),
138          plot.title = element_text(family="GT America", color="#05032d", size=20),
139          plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1),
140          plot.caption = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1, vjust=-3),
141          panel.grid = element_line(color="#ececf0", size=.3),
142          plot.margin=unit(c(1,1,1.5,1.2),"cm"))
143
144
145  ## AMERICA VERSUS ALL OTHER COUNTRIES AND REGIONS
146
```

```r
147  # first we need to build some regex with a vector of country names - use the
     countrycode_data-dataframe in the countrycode-package
148  summary(countrycode_data)
149  unique(countrycode_data$region)
150  countrycode_data$country.name.en
151
152  grepCountriesForBash<-paste0("&& grep -o", " '",
     countrycode_data$country.name.en.regex, "' ", "$file | wc -l | tr -d 'XXX' && printf
     \"YYY\" \\")
153
154  write.table(grepCountriesForBash, file="mypath/analysis/grepCountries.txt",
     row.names=F, quote=F, fileEncoding="UTF-8")
155  ### CAREFUL: REPLACE XXX BY \n IN THE .TXT FILE and YYY BY \t. THEN TRANSFORM INTO
     BASH-SCRIPT stateOfUnion.sh
156  ### CAREFUL 2: THERE WAS A LOT OF HAND-CLEANING AND -AUGMENTING IN THE BASH-SCRIPT
     BELONGING TO THIS
157
158  # see bash-script for text analysis
159
160  # read in countries-per-file, regions-per-file and define colnames
161  countriesPerFile<-read.table("mypath/data/datafiles/countriesPerFileC2.txt", sep="\t")
162  colnames(countriesPerFile)<-c("file", countrycode_data$country.name.en)
163
164  regionsPerFile<-read.table("mypath/data/datafiles/regionsPerFileC2.txt", sep="\t")
165  colnames(regionsPerFile)<-c("file", "asiaPacific", "europe", "africa", "polynesia",
     "caribbean", "southAmerica", "northAmerica", "middleNearEast", "southAsia")
166
167  #replace the US-column in countriesPerFile by the row sums of americPerFile, because
     this was counted in a methodologically cleaner way (corpus 1, including 'US' but not
     'us')
168  americAll<-rowSums(americPerFile[,2:length(americPerFile)])
169  countriesPerFile$`United States of America`<-americAll
170
171  #remove US-column for row-sums without US
172  countriesPerFile$`United States of America`
173  countriesPerFile[,256]
174  countriesPerFile_noUS<-countriesPerFile[,-c(256)]
175
176  #verify
177  countriesPerFile_noUS[,grep("United States", colnames(countriesPerFile_noUS))]
178  head(rowSums(countriesPerFile[,2:length(countriesPerFile)]))
179  head(rowSums(countriesPerFile_noUS[,2:length(countriesPerFile_noUS)]))
180  head(americAll)
181
182  #all countries but US: kick out doubles (2nd and 3rd occurence of yemen, take korea as
     a whole and not north and south korea,
183  #get rid of BRD, DDR and only keep germany as a whole, get rid of austria-hungary, and
     of 2nd occurence of virgin islands)
184  grep("Korea", colnames(countriesPerFile_noUS)) #north korea: 129, south korea: 194
185  grep("German", colnames(countriesPerFile_noUS)) #BRD: 79, DDR: 89
186  grep("Austria", colnames(countriesPerFile_noUS)) #austria-hungary: 17
187  grep("Hungar", colnames(countriesPerFile_noUS))
188  grep("Yemen", colnames(countriesPerFile_noUS)) #2nd and 3rd: 268, 269
189  grep("Virgin", colnames(countriesPerFile_noUS)) #2nd: 263
190
191  countriesPerFile_red<-countriesPerFile_noUS[,-c(17, 79, 89, 129,194,263, 268, 269)]
192
```

```
193  ### plot us vs rest with regions, too
194  countriesRegionsPerFile<-data.frame(countriesPerFile_red, regionsPerFile[,-1])
195  allButUS_regions<-rowSums(countriesRegionsPerFile[,2:length(countriesRegionsPerFile)])
196
197  usVsRest_regions<-data.frame("file"=countriesPerFile$file,
     "USA"=countriesPerFile$`United States of America`, "rest"=allButUS_regions)
198
199  metadata_usVsRest_regions<-metadata %>%
200    bind_cols(usVsRest_regions) %>%
201    group_by(year) %>%
202    mutate(USA_prop=100*(USA/nWords)) %>%
203    mutate(rest_prop=100*(rest/nWords)) %>%
204    select(-USA, -rest) %>%
205    gather(key=partOfWorld, value=prop, -president, -year, -party,-nWords, -nChars, -
     file)
206
207  # plot
208  ggplot(metadata_usVsRest_regions, aes(as.numeric(as.character(year)), prop,
     col=partOfWorld)) +
209    geom_point() +
210    geom_smooth(span=0.1, se=F) +
211    ylab("Anteil an allen Wörtern pro Rede")+
212    theme_minimal() +
213    scale_x_continuous(breaks=seq(1790,2018, 5))+
214    theme(axis.text=element_text(family="GT America", color="#05032d", size=10,
     angle=90),
215         axis.title.x=element_blank(),
216         axis.title.y=element_text(family="GT America", color="#05032d", size=13),
217         plot.title = element_text(family="GT America", color="#05032d", size=20),
218         plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1),
219         plot.caption = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1, vjust=-3),
220         panel.grid = element_line(color="#ececf0", size=.3),
221         plot.margin=unit(c(1,1,1.5,1.2),"cm"))
222  ggsave("usVsRest_pointSmoothe.svg", width = 18, height=9)
223
224
225  ## 20 most named countries overall
226  countryTotals<-colSums(countriesPerFile[,2:length(countriesPerFile)])
227  countryTotals
228  sort(countryTotals, decreasing = T)
229
230  #the 20 most frequent countries
231  frequentCountries<-data.frame(countriesPerFile$file,
232                                countriesPerFile$`United States of America`,
233                                countriesPerFile$Mexico,
234                                countriesPerFile$`United Kingdom of Great Britain and
     Northern Ireland`,
235                                countriesPerFile$Spain,
236                                countriesPerFile$`Russian Federation`,
237                                countriesPerFile$France,
238                                countriesPerFile$China,
239                                countriesPerFile$Cuba,
240                                countriesPerFile$Japan,
241                                countriesPerFile$Germany,
242                                countriesPerFile$Panama,
```

```r
243                                    countriesPerFile$Iraq,
244                                    countriesPerFile$Nicaragua,
245                                    countriesPerFile$`Iran (Islamic Republic of)`,
246                                    countriesPerFile$Korea,
247                                    countriesPerFile$`Viet Nam`,
248                                    countriesPerFile$Brazil,
249                                    countriesPerFile$Canada,
250                                    countriesPerFile$Afghanistan,
251                                    countriesPerFile$Colombia,
252                                    countriesPerFile$Philippines)
253
254   # plot overall frequency
255   frequentCountries_totals <- colSums(frequentCountries[,2:length(frequentCountries)])
256   nbs<-as.numeric(as.character(frequentCountries_totals))
257   nms<-c("USA", "Mexiko", "Grossbritannien", "Spanien", "Russland / Sowjetunion",
      "Frankreich", "China", "Kuba", "Japan", "Deutschland", "Panama", "Irak", "Nicaragua",
      "Iran", "Korea", "Vietnam", "Brasilien", "Kanada", "Afghanistan", "Kolumbien",
      "Philippinen")
258   frequentCountries_totals<-data.frame("country"=nms, "counts"=nbs)
259
260   #reorder levels according to frequency
261   orderVar<-frequentCountries_totals$counts
262   frequentCountries_totals$country<-factor(frequentCountries_totals$country,
      levels=frequentCountries_totals$country[order(orderVar, decreasing=T)])
263   print(levels(frequentCountries_totals$country)) #check
264
265   frequentCountries_noUS<-frequentCountries_totals[grep("USA",
      frequentCountries_totals$country, invert = T),]
266
267   ggplot(frequentCountries_noUS, aes(country, counts))+
268     geom_bar(stat="identity")+
269     ylab("Anzahl Erwähnungen insgesamt")+
270     theme_minimal() +
271     theme(axis.text=element_text(family="GT America", color="#05032d", size=10,
      angle=90),
272          axis.title.x=element_blank(),
273          axis.title.y=element_text(family="GT America", color="#05032d", size=13),
274          plot.title = element_text(family="GT America", color="#05032d", size=20),
275          plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
      hjust=-0.1),
276          plot.caption = element_text(family="GT America", color="#05032d", size=10,
      hjust=-0.1, vjust=-3),
277          panel.grid = element_line(color="#ececf0", size=.3),
278          plot.margin=unit(c(1,1,1.5,1.2),"cm"))
279   ggsave("frequentCountries_bar.svg", width = 18, height=9)
280
281   # plot over time: small mult
282   colnames(frequentCountries)<-c("file", "USA", "Mexiko", "Grossbritannien", "Spanien",
      "Russland / Sowjetunion", "Frankreich", "China", "Kuba", "Japan", "Deutschland",
      "Panama", "Irak", "Nicaragua", "Iran", "Korea", "Vietnam", "Brasilien", "Kanada",
      "Afghanistan", "Kolumbien", "Philippinen")
283
284   # calculate proportions relative to nWords per file
285   metadata_countries<-metadata %>%
286     bind_cols(frequentCountries) %>%
287     group_by(year) %>%
288     summarize_at(vars(USA:Philippinen, nWords),sum) %>%
```

```r
289   mutate_at(vars(USA:Philippinen),funs((. / nWords)*100)) %>%
290   select(-Philippinen, -Kolumbien, -Iran, -Nicaragua, -Panama, -Brasilien)  %>%
291   gather(key=country, value=prop, -year)
292
293 #reorder levels according to frequency of occurence
294 frequentCountries_totals_red<-
    frequentCountries_totals[grep("Philip|Kolum|Iran|Nicara|Panam|Brasi",
    frequentCountries_totals$country, invert=T),]#get rid of unused countries
295
296 metadata_countries_augm<-merge(metadata_countries, frequentCountries_totals_red,
    by.x="country", by.y="country", all=F)
297 metadata_countries_augm$country<-as.factor(metadata_countries_augm$country)
298 levels(metadata_countries_augm$country)
299 metadata_countries_augm$country <- factor(metadata_countries_augm$country,
    levels(metadata_countries_augm$country)[c(14,11,5,13,12,4,2,10,7,3,6,15,9,8,1)])
300
301 #plot small mult
302 ggplot(metadata_countries_augm, aes(as.numeric(as.character(year)),prop, col=country))
    +
303   geom_line() +
304   ylab("Anteil an allen Wörtern pro Rede")+
305   facet_wrap(~country, ncol=5) +
306   theme_minimal() +
307   theme(axis.text=element_text(family="GT America", color="#05032d", size=11),
308         axis.title.x=element_blank(),
309         axis.title.y=element_text(family="GT America", color="#05032d", size=13),
310         plot.title = element_text(family="GT America", color="#05032d", size=20),
311         plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
    hjust=-0.1),
312         plot.caption = element_text(family="GT America", color="#05032d", size=10,
    hjust=-0.1, vjust=-3),
313         panel.grid = element_line(color="#ececf0", size=.3),
314         plot.margin=unit(c(1,1,1.5,1.2),"cm"))
315 ggsave("smallMultiples_desktopTest.svg", plot = last_plot(), width=16, height=9)
316
317
318 #plot one-by-one for small-multiple-layout (y-axis with USA, same for all)
319 for(i in unique(metadata_countries_augm$country)){
320   filename<-paste0(gsub("/", "", i), "Xtime_smallmultiples.svg")
321   countryi<-filter(metadata_countries_augm, country==i)
322   ggplot(countryi, aes(as.numeric(as.character(year)),prop)) +
323     geom_line() +
324     ggtitle(paste0(i))+
325     ylab("Anteil an allen Wörtern pro Rede")+
326     ylim(0,1.78)+ #possibly adapt
327     scale_x_continuous(breaks=c(1790, seq(1800,2010, 50),2018))+
328     theme_minimal() +
329     theme(axis.text=element_text(family="GT America", color="#05032d", size=10,
    angle=90),
330         #axis.title.x=element_text(family="GT America", color="#05032d", size=13,
    hjust=1, vjust=-4), # or play around with something like vjust=-4/nchar(title.chars)
331         axis.title.x=element_blank(),
332         axis.title.y=element_text(family="GT America", color="#05032d", size=13),
333         plot.title = element_text(family="GT America", color="#05032d", size=20), #
    or play around with something like hjust=-0.12*(nchar(indicator)/10)
334         plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
    hjust=-0.1),
```

```r
335        plot.caption = element_text(family="GT America", color="#05032d", size=10,
    hjust=-0.1, vjust=-3),
336            panel.grid = element_line(color="#ececf0", size=.3),
337            plot.margin=unit(c(1,1,1.5,1.2),"cm"))
338    ggsave(filename, width=9, height=9)
339 }
340
341
342 #plot one-by-one for grouped small-multiple-layout (y-axis without USA but same for
    all)
343 metadata_countries_augm_noUS<-metadata_countries_augm[grep("USA",
    metadata_countries_augm$country, invert=T),]
344 for(i in unique(metadata_countries_augm_noUS$country)){
345    filename<-paste0(gsub("/", "", i), "Xtime_groups.svg")
346    countryi<-filter(metadata_countries_augm, country==i)
347    ggplot(countryi, aes(as.numeric(as.character(year)),prop)) +
348      geom_line() +
349      ggtitle(paste0(i))+
350      ylab("Anteil an allen Wörtern pro Rede")+
351      ylim(0,1.25)+ #possibly adapt
352      scale_x_continuous(breaks=c(1790, seq(1800,2010, 50),2018))+
353      theme_minimal() +
354      theme(axis.text=element_text(family="GT America", color="#05032d", size=10,
    angle=90),
355            axis.title.x=element_blank(),
356            axis.title.y=element_text(family="GT America", color="#05032d", size=13),
357            plot.title = element_text(family="GT America", color="#05032d", size=20), #
    or play around with something like hjust=-0.12*(nchar(indicator)/10)
358            plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
    hjust=-0.1),
359            plot.caption = element_text(family="GT America", color="#05032d", size=10,
    hjust=-0.1, vjust=-3),
360            panel.grid = element_line(color="#ececf0", size=.3),
361            plot.margin=unit(c(1,1,1.5,1.2),"cm"))
362    ggsave(filename, width=9, height=9)
363 }
364
365
366 ### ANALYSIS OF THE MOST FREQUENT COUNTRIES PER PRESIDENT ###
367
368 metadata_countries<-metadata %>%
369    bind_cols(countriesPerFile) %>%
370    gather(key=country, value=counts, -president, -year, -party,-nWords, -nChars, -file)
371
372 #get most important countries per president - and mean proportion per country (i.e., we
    value each year the same, regardless of whether there was more or less text)
373 presiCntry<-metadata_countries %>%
374    group_by(president, country) %>%
375    summarise(cntryMean=n(counts), startYear=min(as.numeric(as.character(year)))) %>%
376    arrange(desc(cntryMean))
377
378 maxCntriesPresi<-data.frame()
379 for(i in unique(metadata_countries$president)){
380    presi<-presiCntry %>% filter(president==i)
381    maxima<-presi[1:6,]
382    maxCntriesPresi<-bind_rows(maxCntriesPresi, maxima)
383 }
```

```
384
385  #get most important countries per president - and sum per country
386  presiCntry_counts<-metadata_countries %>%
387    group_by(president, country) %>%
388    summarise(cntrySum=sum(counts), startYear=min(as.numeric(as.character(year)))) %>%
389    arrange(desc(cntrySum))
390
391  #get rid of doubles
392  presiCntry_counts_red<-presiCntry_counts[grep("Heard|Federal|Democratic",
     presiCntry_counts$country, invert=T),]
393  print(presiCntry_counts_red[presiCntry_counts_red$president=="donald j trump",], n=25)
394
395  #with sums, not means
396  maxCntriesPresi<-data.frame()
397  for(i in unique(metadata_countries$president)){
398    presi<-presiCntry_counts_red %>% filter(president==i)
399    maxima<-presi[1:8,]
400    maxCntriesPresi<-bind_rows(maxCntriesPresi, maxima)
401  }
402
403  #reorder factor levels according to startYear of president
404  maxCntriesPresi$startYear<-as.numeric(as.character(maxCntriesPresi$startYear)) #  adapt
     to the reuired variable
405  maxCntriesPresi$president<-factor(maxCntriesPresi$president,
     levels(maxCntriesPresi$president)[order(unique(maxCntriesPresi$startYear),
     decreasing=F)])
406  print(levels(maxCntriesPresi$president)) #check
407
408  # plot
409  ggplot(maxCntriesPresi, aes(country,cntrySum)) +
410    geom_bar(stat="identity", fill="lightgray") +
411    facet_wrap(~president, scales="free") +
412    ggtitle("Andere Länder in den Reden der US-Präsidenten")+
413    ylab("Proportional zur Anzahl Wörter in Rede, in Prozent")+
414    geom_text(aes(label=country, x=country, y=(cntrySum-cntrySum)), hjust=-.3, angle=90,
     family="GT America", color="#05032d", size=2.5)+
415    theme_minimal() +
416    theme(axis.text.y=element_text(family="GT America", color="#05032d", size=11),
417          axis.text.x=element_blank(),
418          axis.title.x=element_blank(),
419          axis.title.y=element_text(family="GT America", color="#05032d", size=13),
420          plot.title = element_text(family="GT America", color="#05032d", size=20),
421          plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1),
422          plot.caption = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1, vjust=-3),
423          panel.grid = element_line(color="#ececf0", size=.3),
424          plot.margin=unit(c(1,1,1.5,1.2),"cm"))
425
426
427  ## Trump
428  trump<-filter(maxCntriesPresi, president =="donald j trump")
429
430  #rename
431  trump$country<-c("USA", "Nordkorea", "China", "Iran", "Mexiko", "Kanada",
     "Afghanistan", "Israel")
432
```

```r
433  #reorder
434  orderVar<-trump$cntrySum
435  trump$country<-factor(trump$country, levels=trump$country[order(orderVar,
     decreasing=T)])

436
437  #plot
438  ggplot(trump, aes(country,cntrySum)) +
439    geom_bar(stat="identity", fill="lightgray") +
440    ggtitle("Donald J. Trump")+
441    ylab("Anzahl Erwähnungen")+
442    theme_minimal() +
443    theme(axis.text.y=element_text(family="GT America", color="#05032d", size=11),
444          axis.text.x=element_text(family="GT America", color="#05032d", size=11,
     angle=90),
445          axis.title.x=element_blank(),
446          axis.title.y=element_text(family="GT America", color="#05032d", size=13),
447          plot.title = element_text(family="GT America", color="#05032d", size=20),
448          plot.subtitle = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1),
449          plot.caption = element_text(family="GT America", color="#05032d", size=10,
     hjust=-0.1, vjust=-3),
450          panel.grid = element_line(color="#ececf0", size=.3),
451          plot.margin=unit(c(1,1,1.5,1.2),"cm"))
452  ggsave("trump.svg")
453
454
```