MSD 2019 Final Project

A replication and extension of Ethnicity, Insurgency, and Civil War by James D. Fearon & David D. Laitin, American Political Science Review

Preston Bradham (PMB2164), Chaim Eisenbach (CE2388), Aysha Khan (ASK2256)

2019-05-12 15:13:20

Contents

Paper Overview

Fearon and Laitin analyzed post-WWII global civil wars to see if they could create a model to predict the likelihood of civil war onset. They investigated unique characteristics of the countries at hand to see if they could hypothesize which variable played the biggest role in the making of the civil war. Such as poverty, political instability, ethnic and religious diversity.

They defined civil war as conflicts that meet three criteria: involved fighting between agents of a state and organized nonstate groups who sought to take control of a government policicies, the conflict killed at least 1,000 over its course with at least an average of 100 yearly deaths, and at least 100 were killed on both sides, including civilians attacked by rebels

Eleven hypotheses were were investigated. For example one was: "measures of country's ethnic or religious diversity should be associated with a higher risk of civil war." And most of the other followed suit but with different parameters in the hypothesis.

The paper concluded that there was not enough evidence to conclude that any of the hypothesis were on target and finished with saying that civil wars are incredible hard to predict, but it is easier to predict insurgencies. It includes a cast of doubt on three wide-held notions concerning political conflict findings: prevalence of civil war in the 1990s was not due to the end of the Cold War, greater religious and/or ethnic

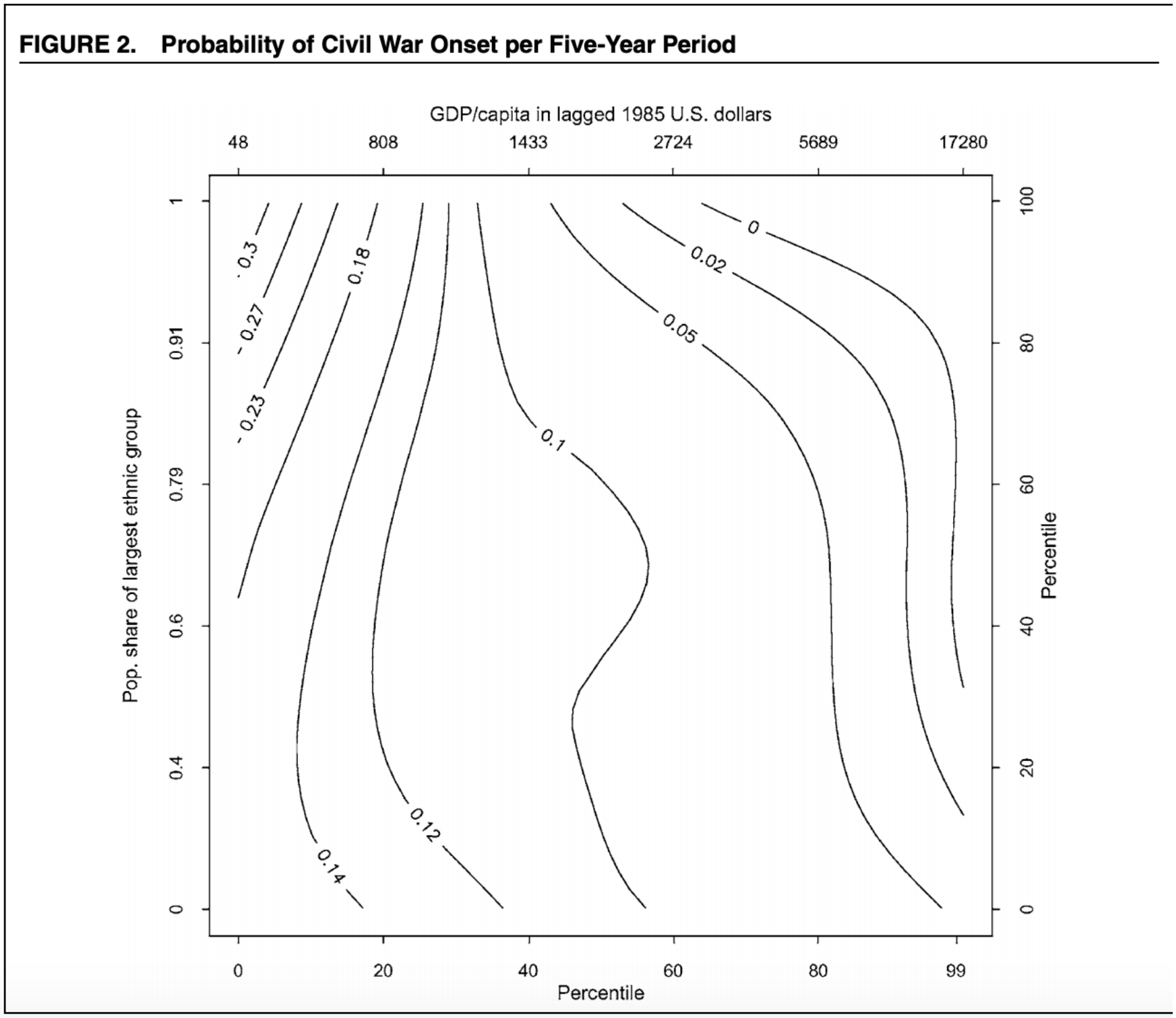


Figure 1: Figure 1

diversity, on its own does not make a country more prone to civil war, and cannot predict where a civil war will break out - based off of strong ethnic or political grievances.

Figures

We seek to replicate the following two figures from the paper. Figure 1 plots the number of countries with ongoing civil wars by year, from 1945 to 1999 (solid line). The paper also shows the proportion of countries with at least one ongoing war in each year (dashed line). What is interesting is that this graph indicates that post-1990s civil wars were not due to the effects of the fall of the Berlin Wall (which signified the end of the Cold War). However, conflicts associated with the fall of the Soviet Union were partly responsible for the sharp increase we witness in the early 1990s.

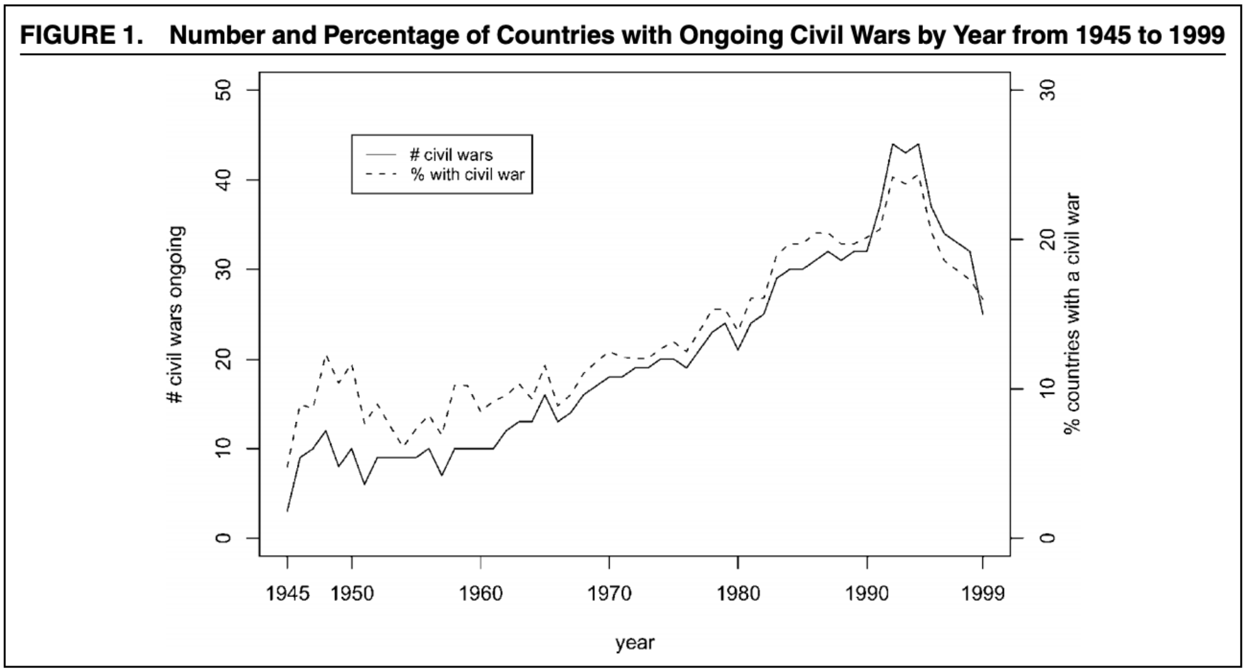


Figure 2: Figure 2

Overview of Data

Their dataset uses data across the world from the period of 1945-1999 on 161 countries that had a population of at least half a million in 1990.

It includes information on the contries: economy, location, population, employment, minerals/resources, civil war information (time frame, deaths, leader, etc.), ethnic onset, oil, GDP, Colonial country, religion percentage, and more. It allows us to explore various parameteres. Originally we had used the data as is, but realized that one row had onset==4 so we got rid of it as it seemed to be an anomaly so we got rid of it.

Conclusions

The paper concludes: "The prevalence of internal war in the 1990s is mainly the result of an accumulation of protracted conflicts since the 1950s rather than a sudden change associated with a new, post-Cold War international system."

Our conclusion for the replication of Figures 1&2 (AYSHA)

Our conclusion for the replication of first column of table 1

Model 1 in Table 1 shows the results of a logit analysis. using onset as the dependent variable. With a standard glm model we were easily able to recreate column 1 of table 1.

Our conclusion for the extensions

By Continent

[Note: South Africa here refers to the geographic South Africa, not specifically the country South Africa] The reason we wanted to see how the civil wars broke down by continent is that we could more accurately use a historical events timeline. So we know that the end of a Cold War was not as important an event as people originally claim. The Cold War ended in ~1991, and you can see from the figure it was in fact South

Africa region that has a increase along with Asia, but eastern European actually saw a decrease after that time period. Before 1960 the civil wars was dominated by Asia and Eastern Europe (which historically is accurate). Then the colonial civil wars started to take place (North Africa and Middle East/ South Africa), and consistently played a key role in the overall average of the civil wars in the world.

We can see that a short time after World War II the only two regions that had civil wars were Eastern Europe and Asia, which complety makes sense since Asia was still deveoloping (from colonialism). And Eastern Europe was engulfed by the USSR so there was a tension between the Soviet Union leaders and the satellite states, and by extension the political leaders withing those states, those loyal to the party and those who were not.

We can look at a these conflicts through an economic view. We know that after World War II the victors had an economic growth period called the postwar economic boom starting at 1950 which lasted until early 1970's. So the countries that were involved in the postwar economic boom saw fewer civil wars during that time period. However, 1971 was the collapse of the Bretton Woods Monetary system, then in 1973 there was an oil crisis, followed by the american economic recession from 1973-1975. These events could suggest why we saw a increase in South African and North Africa/Middle East civil wars, as the vestiges of institutional colonialism was fading away and the oil crisis brought a downturn to their economy which drove them to civil wars. This is similar to the political conflicts in European states post WWI, where there was a civil war.

Then with the collapse of the Soviet Union in late 1991 brought some civil wars to East Europe and Asia, but from what we can see it was not a true indicator of civil wars (from the percentages).

We believe that the rise of the South Africa civil wars in the 90's(after the colonial revolutions) was due to mineral resources. Internal conflict was brought up due to different groups of a recently developed country fighting for minerals in order to get money.

Predictability of model

We dropped one Russia becase it was the only country whose onset (which is what the paper is trying to predict on) is not 0 or 1.

Figure 1 Replication

```r
repdata <- read.dta("./data/repdata.dta")
# removing onset == 4
repdata <-repdata[-2496, ]

glance(repdata)
```

```
## Warning: 'glance.data.frame' is deprecated.
## See help("Deprecated")
```

```
## # A tibble: 1 x 4
##    nrow  ncol complete.obs na.fraction
##   <int> <int>        <int>       <dbl>
## 1  6609    69           69      0.0765
```

```r
# View(repdata)

sumwars_per_year <- repdata %>%
  group_by(year) %>%
  filter(war == 1) %>%
  summarize(
    count_wars_total = sum(wars)
  )
```

```
wars_per_year <- repdata %>%
  group_by(year) %>%
  filter(war == 1) %>%
  summarize(
    count_wars = sum(war)
  )

raw_num_countries <- repdata %>%
  group_by(year) %>%
  summarize(
    count_countries = sum(n())
  ) %>%
  ungroup(year)

perc_civil_war <- merge(wars_per_year, raw_num_countries, by = "year")
perc_civil_war <- merge(perc_civil_war, sumwars_per_year, by = "year")

# View(perc_civil_war)

perc_civil_war$perc <- (perc_civil_war$count_wars/perc_civil_war$count_countries)*100

plot(perc_civil_war$year, perc_civil_war$count_wars_total, axes = FALSE,
     ylim = c(0, 50), xlab = "", ylab = "", type = "l",
     col = "blue", main = "# and % of Countries with Ongoing Civil Wars (1945-1999)")
axis(2, ylim = c(0, 50), col = "black", las = 1)
mtext("# Ongoing Wars", side = 2, col = "black", line = 2.5)

# Plot the second plot and draw the axis on the right
par(new = TRUE)
plot(perc_civil_war$year, perc_civil_war$perc, pch = "solid", xlab = "", ylab = "", ylim = c(0, 30), axe
mtext("% Countries with a Civil War", side = 4, col = "black", line = 2.5)
axis(4, ylim = c(0, 30), col = "black", col.axis = "black")

# Draw the time axis
axis(1, pretty(range(perc_civil_war$year), 4))
mtext("Year", side = 1, col = "black", line = 2.5)

# Draw the legend
legend("topleft", legend = c("# civil wars", "% with civil war"),
       text.col = c("blue", "red"), col = c("blue", "red"), lty = 1:2, cex = 0.7)
```

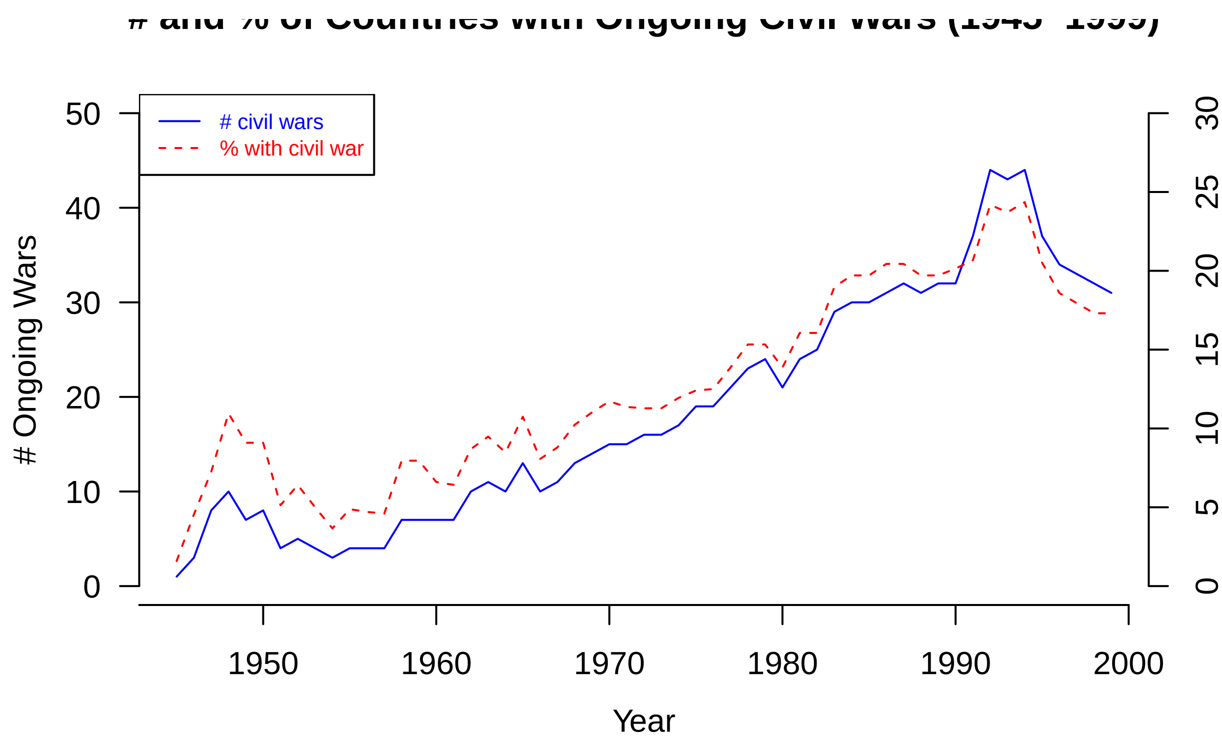


Figure 2 Replication: Part A

```r
gdp_per_year <- repdata %>%
  drop_na(gdpen) %>%
  drop_na(pop) %>%
  group_by(year) %>%
  summarize(
    gdp_sum = sum(gdpen),
    pop_sum = sum(pop)
  )

gdp_per_year$gdp_pc <- gdp_per_year$gdp_sum / gdp_per_year$pop_sum * 10000

gdp_per_year_perc <- merge(gdp_per_year, perc_civil_war, by = "year")
gdp_per_year_perc$cv_percentile <- round(gdp_per_year_perc$perc / max(gdp_per_year_perc$perc), digits =

# View(gdp_per_year_perc)

gdp_per_year_perc %>%
  ggplot(aes(x = perc, y = gdp_pc)) +
  geom_line(color = "red")
```

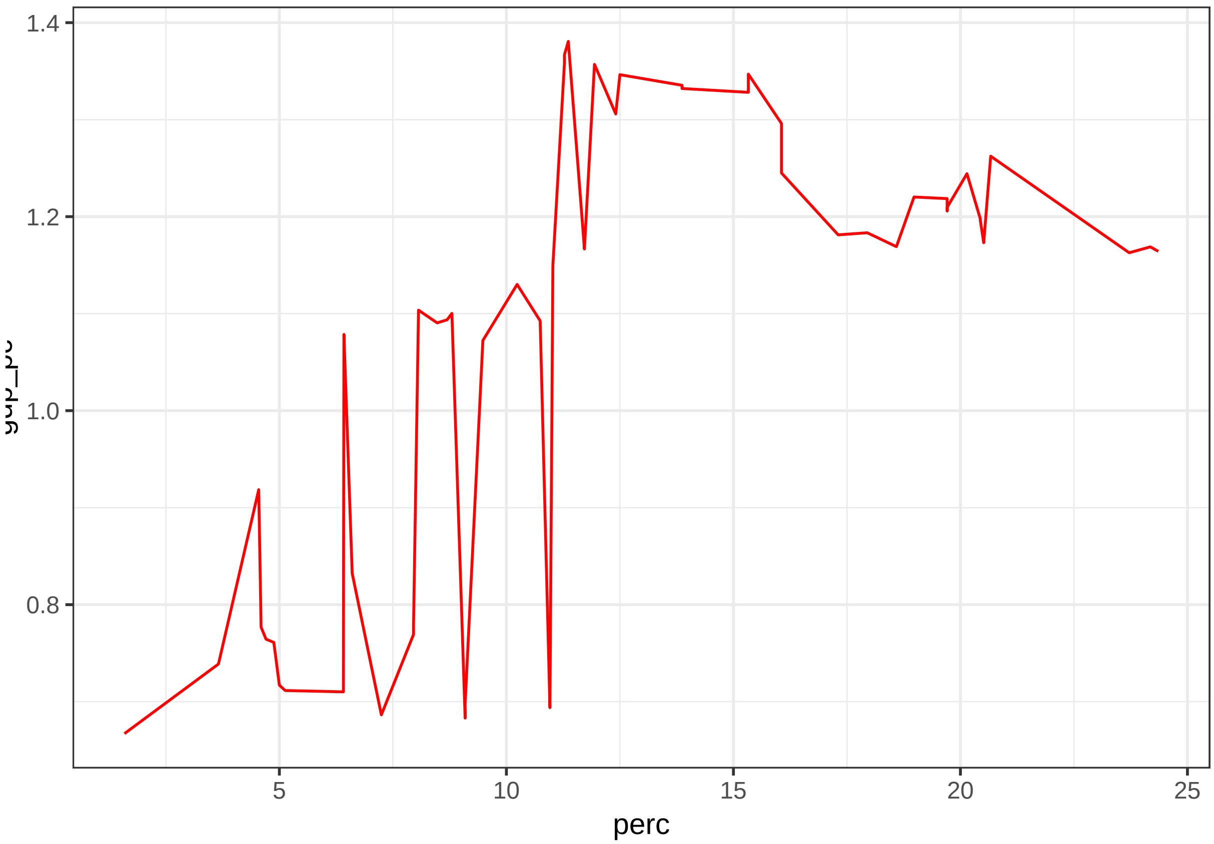
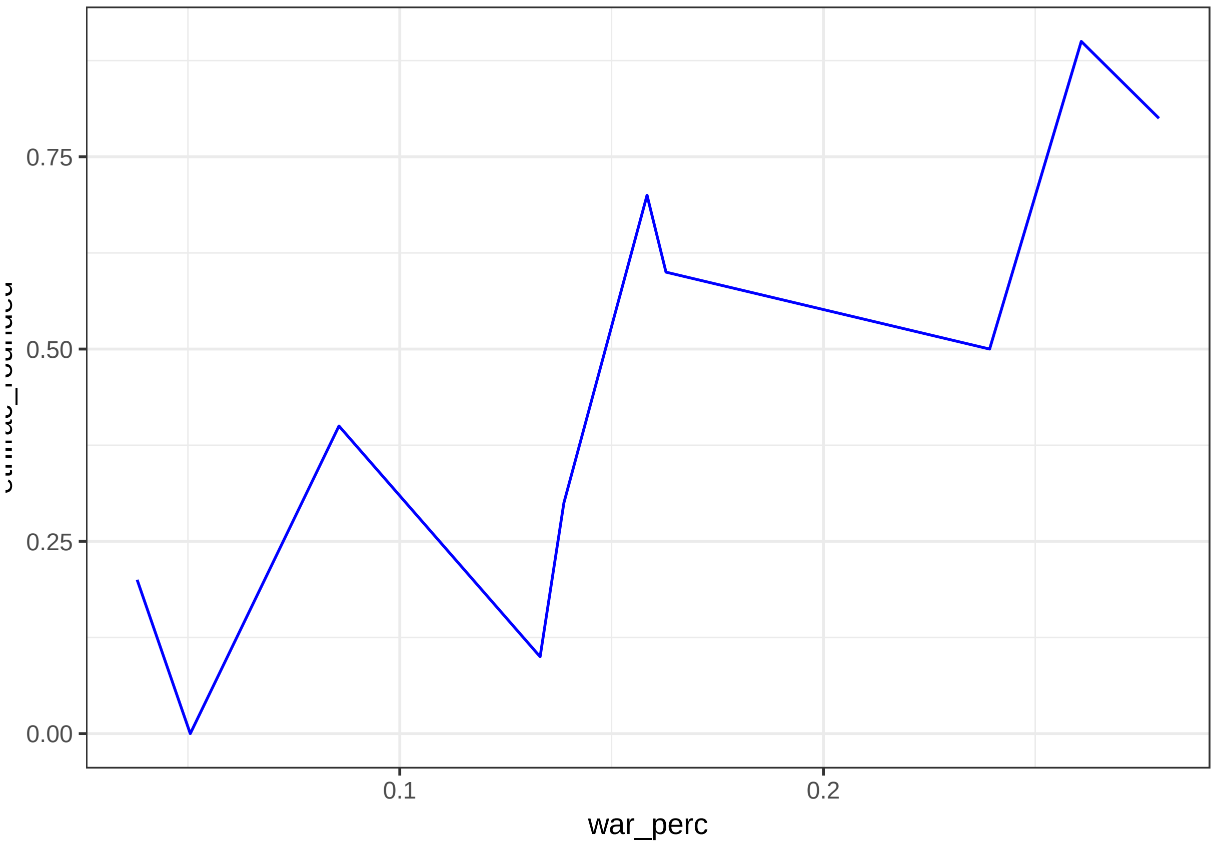


Figure 2 Replication: Part B

```
rep_data_eth <- repdata %>%
  select(ethfrac, war) %>%
  mutate(ethfrac_rounded = round(ethfrac, digits = 1)) %>%
  group_by(ethfrac_rounded) %>%
  summarize(
    sum_countries = sum(n()),
    sumwars_per_eth = sum(war)
  )

rep_data_eth$war_perc = rep_data_eth$sumwars_per_eth/rep_data_eth$sum_countries

# View(rep_data_eth)

rep_data_eth %>%
  ggplot(aes(x = war_perc, y = ethfrac_rounded)) +
  geom_line(color = "blue")
```




```r
# using everything the paper does for table 1
mylogit1 <- glm(onset ~ warl + gdpenl + lpopl1 + lmtnest
+ ncontig + Oil + nwstate + instab + polity2l + ethfrac + relfrac,  data =repdata, family = "binomial")
summary(mylogit1)
```

```
##
## Call:
## glm(formula = onset ~ warl + gdpenl + lpopl1 + lmtnest + ncontig +
##     Oil + nwstate + instab + polity2l + ethfrac + relfrac, family = "binomial",
##     data = repdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1298  -0.1998  -0.1446  -0.1009   3.4131
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.66554    0.73917  -9.018  < 2e-16 ***
## warl        -0.92448    0.31432  -2.941 0.003270 **
## gdpenl      -0.34659    0.07244  -4.785 1.71e-06 ***
## lpopl1       0.25650    0.07314   3.507 0.000453 ***
## lmtnest      0.22054    0.08488   2.598 0.009367 **
## ncontig      0.39191    0.27733   1.413 0.157615
## Oil          0.88587    0.27942   3.170 0.001522 **
## nwstate      1.71739    0.33858   5.072 3.93e-07 ***
## instab       0.62541    0.23554   2.655 0.007926 **
## polity2l     0.02353    0.01681   1.400 0.161656
```

```
## ethfrac      0.14435    0.37490   0.385 0.700211
## relfrac      0.28516    0.51072   0.558 0.576606
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1068.92  on 6325  degrees of freedom
## Residual deviance:  954.44  on 6314  degrees of freedom
##   (283 observations deleted due to missingness)
## AIC: 978.44
##
## Number of Fisher Scoring iterations: 8
```

```r
# removing gdp
mylogit2<- glm(onset ~ warl  + lpopl1 + lmtnest
+ ncontig + Oil + nwstate + instab + polity2l + ethfrac + relfrac,  data =repdata, family = "binomial")
summary(mylogit2)
```

```
##
## Call:
## glm(formula = onset ~ warl + lpopl1 + lmtnest + ncontig + Oil +
##     nwstate + instab + polity2l + ethfrac + relfrac, family = "binomial",
##     data = repdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.8056  -0.1923  -0.1486  -0.1162   3.3210
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.61377    0.72549 -10.495  < 2e-16 ***
## warl        -0.65890    0.30503  -2.160  0.03076 *
## lpopl1       0.22018    0.07590   2.901  0.00372 **
## lmtnest      0.26488    0.08316   3.185  0.00145 **
## ncontig      0.27558    0.27149   1.015  0.31008
## Oil          0.46551    0.26050   1.787  0.07393 .
## nwstate      2.14401    0.31204   6.871 6.38e-12 ***
## instab       0.91537    0.22980   3.983 6.79e-05 ***
## polity2l    -0.01855    0.01527  -1.215  0.22444
## ethfrac      0.86025    0.37592   2.288  0.02211 *
## relfrac      0.22680    0.49024   0.463  0.64363
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1108.2  on 6524  degrees of freedom
## Residual deviance: 1023.1  on 6514  degrees of freedom
##   (84 observations deleted due to missingness)
## AIC: 1045.1
##
## Number of Fisher Scoring iterations: 7
```

```r
# removing ethfrac
mylogit3 <- glm(onset ~ warl  + gdpenl + lpopl1 + lmtnest
+ ncontig + Oil + nwstate + instab + polity2l +  relfrac,  data =repdata, family = "binomial")
summary(mylogit3)
```

```
##
## Call:
## glm(formula = onset ~ warl + gdpenl + lpopl1 + lmtnest + ncontig +
##     Oil + nwstate + instab + polity2l + relfrac, family = "binomial",
##     data = repdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1252  -0.1994  -0.1448  -0.1008   3.4184
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.63112    0.73057  -9.077  < 2e-16 ***
## warl        -0.91248    0.31239  -2.921 0.003490 **
## gdpenl      -0.35349    0.07052  -5.012 5.38e-07 ***
## lpopl1       0.25951    0.07217   3.596 0.000323 ***
## lmtnest      0.21710    0.08441   2.572 0.010116 *
## ncontig      0.39448    0.27634   1.428 0.153426
## Oil          0.90797    0.27349   3.320 0.000900 ***
## nwstate      1.72288    0.33833   5.092 3.54e-07 ***
## instab       0.62620    0.23549   2.659 0.007833 **
## polity2l     0.02372    0.01680   1.412 0.157902
## relfrac      0.33385    0.49474   0.675 0.499802
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1068.92  on 6325  degrees of freedom
## Residual deviance:  954.58  on 6315  degrees of freedom
##   (283 observations deleted due to missingness)
## AIC: 976.58
##
## Number of Fisher Scoring iterations: 8
```

Analysis using Stan and Loo

```r
# using everything the paper does for table 1
# resource https://uw-csss-564.github.io/assignment-2017-4//
#mylogit1_stan <- stan_glm(onset ~ warl + gdpenl + lpopl1 + lmtnest
#+ ncontig + Oil + nwstate + instab + polity2l + ethfrac + relfrac,  data =repdata, family = "binomial")
#summary(mylogit1)
# removing gdp
#mylogit2_stan<- stan_glm(onset ~ warl  + lpopl1 + lmtnest
#+ ncontig + Oil + nwstate + instab + polity2l + ethfrac + relfrac,  data =repdata, family = "binomial")
#summary(mylogit2)

# Leave-One-Out (LOO) cross-validation, which is implemented by the loo function in the loo package
```

```
#loo_mod1 <- loo(mylogit1_stan)
#loo_mod2 <- loo(mylogit2_stan)
#compare(loo_mod1,loo_mod2)
```

It took too long to run, so we commented it out becuase we kept on having to run our code. But it does show that the model that doesn't have gdp is slightly worse, which is of course expeced.

Some more validations on the logit model

```
data(repdata)
```

```
## Warning in data(repdata): data set 'repdata' not found
```

```
Train <- createDataPartition(repdata$onset, p=0.6, list=FALSE)
training <- repdata[ Train, ]
testing <- repdata[ -Train, ]

mylogit1_test <- glm(as.factor(onset) ~ warl  + lpopl1 + lmtnest
+ ncontig + Oil + nwstate + instab + polity2l + ethfrac + relfrac,  data = training,family = "binomial")
mylogit1_train_pred  <- predict(mylogit1_test, data=training,type="response")
head(mylogit1_train_pred)
```

```
##          1          3          4          6          7          8
## 0.02853555 0.02857677 0.02860973 0.02882976 0.02900402 0.02911287
```

```
mylogit1_test_pred <- predict(mylogit1_test, data=testing,type="response")
head(mylogit1_test_pred)
```

```
##          1          3          4          6          7          8
## 0.02853555 0.02857677 0.02860973 0.02882976 0.02900402 0.02911287
```

```
log_odds = predict(mylogit1, testing)
```

conclusion for the replication of first column of table 1

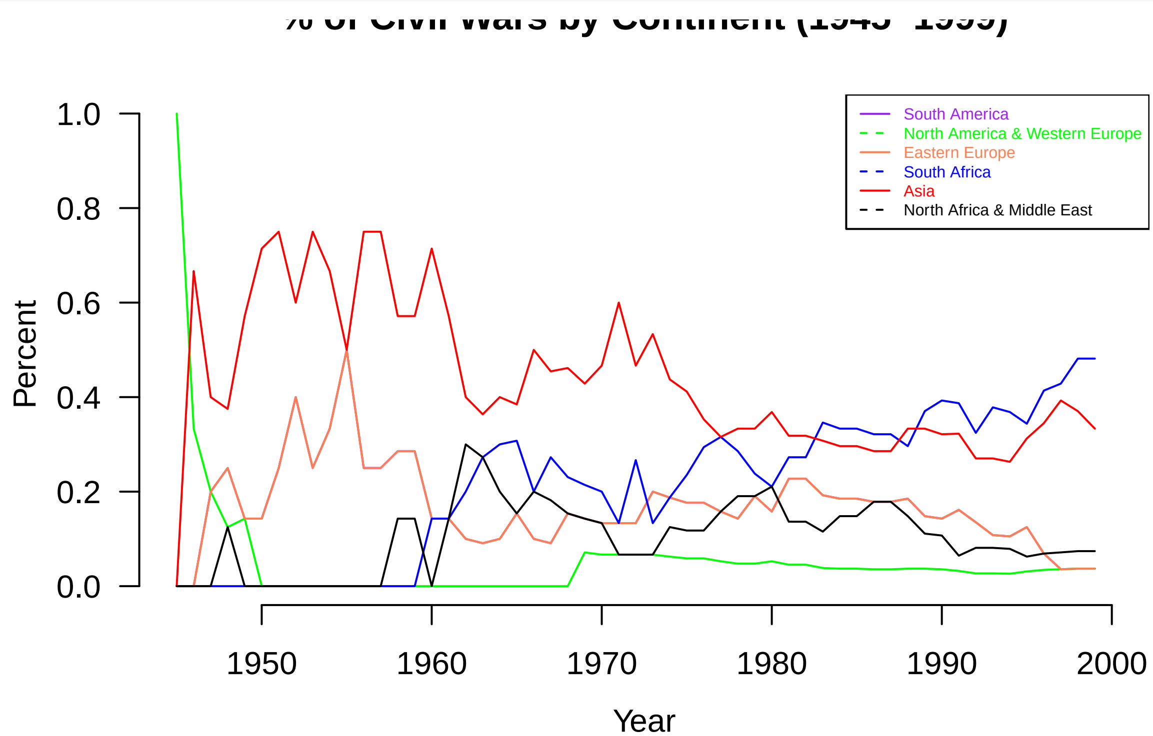
Model 1 in Table 1 shows the results of a logit analysis using onset as the dependent variable.

Extensions

```
#First five lines filter the oringal data (repdata) and such that we create the total number of civil w
#then the number of wars per year
# then just the number of countries,
#from this we can create the perc_civil_war where it is the same as figure2





sumwars_per_year <- repdata %>%
  group_by(year) %>%
  filter(war == 1) %>%
  summarize(
    count_wars_total = sum(wars)
  )

wars_per_year <- repdata %>%
```

```r
  group_by(year) %>%
  filter(war == 1) %>%
  summarize(
    count_wars = sum(war)
  )

raw_num_countries <- repdata %>%
  group_by(year) %>%
  summarize(
    count_countries = sum(n())
  ) %>%
  ungroup(year)

perc_civil_war <- merge(wars_per_year, raw_num_countries, by = "year")
perc_civil_war <- merge(perc_civil_war, sumwars_per_year, by = "year")

#We then want to filter the original by region: South America, Western, East Europe,
#South Africa, Asia, and North Africa/Middle East
#Then take the total of civil wars per for each region
# then normalize them as percents by dividing each by the perc_civil_war from above


#South America
southamerica <- repdata  %>% filter (lamerica == 1)
wars_per_year_SA <- southamerica %>% group_by( year) %>% summarize(count_wars = sum(war))
percent_SA <- (wars_per_year_SA$count_wars/perc_civil_war$count_wars)

#Western
western <- repdata %>% filter (western == 1)
wars_per_year_WS <- western %>% group_by(year)  %>% summarize(count_wars = sum(war))
percent_WS <- (wars_per_year_WS$count_wars/perc_civil_war$count_wars)

#East Europe
easteurope <- repdata  %>% filter (eeurop == 1)
wars_per_year_EE <- easteurope %>% group_by(year)  %>% summarize(count_wars = sum(war))
percent_EE <- (wars_per_year_SA$count_wars/perc_civil_war$count_wars)

#South Africa
southafrica <- repdata  %>% filter (ssafrica == 1)
wars_per_year_SAF <- southafrica%>% group_by(year)  %>% summarize(count_wars = sum(war))
percent_SAF <- (wars_per_year_SAF$count_wars/perc_civil_war$count_wars)

#Asia
asia <- repdata  %>% filter (asia == 1)
wars_per_year_AS <- asia %>% group_by(year) %>% summarize(count_wars = sum(war))
percent_AS <- (wars_per_year_AS$count_wars/perc_civil_war$count_wars)

#North African and Middle East
northafricamiddleeast <- repdata %>% filter (nafrme == 1)
wars_per_year_NAM <- northafricamiddleeast %>% group_by(year) %>% summarize(count_wars = sum(war))
percent_NAM <- (wars_per_year_NAM$count_wars/perc_civil_war$count_wars)
```

```
#Plot each continent percent civil war on same plot to show differences
plot(wars_per_year_SA$year, percent_SA, axes = FALSE,
     ylim = c(0, 1), xlim = c(1945, 2000), xlab = "" , ylab = "", type = "l",
     col = "purple", main = "% of Civil Wars by Continent (1945-1999)")
lines(wars_per_year_WS$year, percent_WS, col = "green")
lines(wars_per_year_EE$year, percent_EE, col = "coral")
lines(wars_per_year_SAF$year, percent_SAF, col = "blue")
lines(wars_per_year_AS$year, percent_AS, col = "red")
lines(wars_per_year_NAM$year, percent_NAM, col = "black")

#Label plot x/y axis
axis(2, ylim = c(0, 1), col = "black", las = 1)
axis(1, xlim = c(1945, 1999), col = "black", las = 1)
mtext("Percent ", side = 2, col = "black", line = 2.5)
mtext("Year", side = 1, col = "black", line = 2.5)

#Give a legend to plot for each continent
legend("topright", legend = c("South America", "North America & Western Europe", "Eastern Europe","Sout
```




```
#We wanted to see each country that had a civil war in each continent and then the time frame for it

#we filtered by country and year looking countries that had civil wars
#ordered it by year, then use ave/paste0  to find the time frame of each civil
#then took away duplicates
#Its important to note that if a country had multiple civil wars over various time frames, we took the
#and the end date of the last for the time frames. This is because there was still internal disruptiond
#Which is why Russia is shown as 1946-1999, becasue they had their internal disputes early on 1946-1950
#up until the contry fell of which the 1992-1999 civil wars started

#South America
```

```r
wars_country_year_SA <- southamerica %>% group_by(country,year) %>% filter(war==1 )%>% summarize(Year =
wars_country_year_SA <- wars_country_year_SA [order(wars_country_year_SA$Year),]
wars_country_year_SA$min = ave(wars_country_year_SA$Year, wars_country_year_SA$country, FUN = min)
wars_country_year_SA$max = ave(wars_country_year_SA$Year, wars_country_year_SA$country, FUN = max)
wars_country_year_SA$range = paste0(wars_country_year_SA$min, " - ", wars_country_year_SA$max)
wars_country_year_SA = wars_country_year_SA[!duplicated(wars_country_year_SA$country),]
wars_country_year_SA <- wars_country_year_SA[,c(1,6)]
names(wars_country_year_SA) <- c("South America", "Conflict Time Frame")


#Western Countries
wars_country_year_W <- western %>% group_by(country,year) %>% filter(war==1 )%>% summarize(Year = year)
wars_country_year_W <- wars_country_year_W [order(wars_country_year_W$Year),]
wars_country_year_W$min = ave(wars_country_year_W$Year, wars_country_year_W$country, FUN = min)
wars_country_year_W$max = ave(wars_country_year_W$Year, wars_country_year_W$country, FUN = max)
wars_country_year_W$range = paste0(wars_country_year_W$min, " - ", wars_country_year_W$max)
wars_country_year_W = wars_country_year_W[!duplicated(wars_country_year_W$country),]
wars_country_year_W <- wars_country_year_W[,c(1,6)]
names(wars_country_year_W) <- c("Western Countries", "Conflict Time Frame")


#East Europe
wars_country_year_EE <- easteurope %>% group_by(country,year) %>% filter(war==1 )%>% summarize(Year = ye
wars_country_year_EE <- wars_country_year_EE [order(wars_country_year_EE$Year),]
wars_country_year_EE$min = ave(wars_country_year_EE$Year, wars_country_year_EE$country, FUN = min)
wars_country_year_EE$max = ave(wars_country_year_EE$Year, wars_country_year_EE$country, FUN = max)
wars_country_year_EE$range = paste0(wars_country_year_EE$min, " - ", wars_country_year_EE$max)
wars_country_year_EE = wars_country_year_EE[!duplicated(wars_country_year_EE$country),]
wars_country_year_EE <- wars_country_year_EE[,c(1,6)]
names(wars_country_year_EE) <- c("East Europe", "Conflict Time Frame")


#South Africa
wars_country_year_SAF <- southafrica %>% group_by(country,year) %>% filter(war==1 )%>% summarize(Year =
wars_country_year_SAF <- wars_country_year_SAF [order(wars_country_year_SAF$Year),]
wars_country_year_SAF$min = ave(wars_country_year_SAF$Year, wars_country_year_SAF$country, FUN = min)
wars_country_year_SAF$max = ave(wars_country_year_SAF$Year, wars_country_year_SAF$country, FUN = max)
wars_country_year_SAF$range = paste0(wars_country_year_SAF$min, " - ", wars_country_year_SAF$max)
wars_country_year_SAF = wars_country_year_SAF[!duplicated(wars_country_year_SAF$country),]
wars_country_year_SAF <- wars_country_year_SAF[,c(1,6)]
names(wars_country_year_SAF) <- c("South Africa", "Conflict Time Frame")


#Asia
wars_country_year_A <- asia %>% group_by(country,year) %>% filter(war==1 )%>% summarize(Year = year)
wars_country_year_A <- wars_country_year_A [order(wars_country_year_A$Year),]
wars_country_year_A$min = ave(wars_country_year_A$Year, wars_country_year_A$country, FUN = min)
wars_country_year_A$max = ave(wars_country_year_A$Year, wars_country_year_A$country, FUN = max)
wars_country_year_A$range = paste0(wars_country_year_A$min, " - ", wars_country_year_A$max)
wars_country_year_A = wars_country_year_A[!duplicated(wars_country_year_A$country),]
wars_country_year_A <- wars_country_year_A[,c(1,6)]
names(wars_country_year_A) <- c("Asia", "Conflict Time Frame")
```

```r
#North Africa and Middle East
wars_country_year_NA <- northafricamiddleeast %>% group_by(country,year) %>% filter(war==1 )%>% summari
wars_country_year_NA <- wars_country_year_NA [order(wars_country_year_NA$Year),]
wars_country_year_NA$min = ave(wars_country_year_NA$Year, wars_country_year_NA$country, FUN = min)
wars_country_year_NA$max = ave(wars_country_year_NA$Year, wars_country_year_NA$country, FUN = max)
wars_country_year_NA$range = paste0(wars_country_year_NA$min, " - ", wars_country_year_NA$max)
wars_country_year_NA = wars_country_year_NA[!duplicated(wars_country_year_NA$country),]
wars_country_year_NA <- wars_country_year_NA[,c(1,6)]
names(wars_country_year_NA) <- c("North Africa and Middle East", "Conflict Time Frame")



#Prints all tables for the Region
formattable(wars_country_year_SA,  align = c("l", rep("r", NCOL(wars_country_year_SA) - 1)))
```

South America

Conflict Time Frame

PARAGUAY

1947 - 1947

COLOMBIA

1948 - 1999

COSTARICA

1948 - 1948

BOLIVIA

1952 - 1952

ARGENTINA

1955 - 1977

CUBA

1958 - 1959

DOMINICAN REP.

1965 - 1965

GUATEMALA

1968 - 1996

NICARAGUA

1978 - 1988

EL SALVADOR

1979 - 1992

PERU

1981 - 1995

HAITI

1991 - 1995

```r
formattable(wars_country_year_W,  align = c("l", rep("r", NCOL(wars_country_year_W) - 1)))
```

Western Countries

Conflict	Time Frame
GREECE	1945 - 1949
UK	1969 - 1999

```r
formattable(wars_country_year_EE,  align = c("l", rep("r", NCOL(wars_country_year_EE) - 1)))
```

East Europe

Conflict	Time Frame
RUSSIA	1947 - 1999
YUGOSLAVIA	1991 - 1991
AZERBAIJAN	1992 - 1994
BOSNIA	1992 - 1995
CROATIA	1992 - 1995
GEORGIA	1992 - 1994
MOLDOVA	1992 - 1992
TAJIKISTAN	1992 - 1997

```r
formattable(wars_country_year_SAF,  align = c("l", rep("r", NCOL(wars_country_year_SAF) - 1)))
```

South Africa

Conflict	Time Frame
DEM. REP. CONGO	1960 - 1999
RWANDA	1962 - 1999
SUDAN	1963 - 1999

CHAD

1965 - 1999

NIGERIA

1967 - 1970

BURUNDI

1972 - 1999

ZIMBABWE

1972 - 1987

ETHIOPIA

1974 - 1999

ANGOLA

1975 - 1999

MOZAMBIQUE

1976 - 1995

SOMALIA

1981 - 1999

UGANDA

1981 - 1999

SOUTH AFRICA

1983 - 1994

LIBERIA

1989 - 1996

MALI

1989 - 1994

SENEGAL

1989 - 1999

SIERRA LEONE

1991 - 1999

DJIBOUTI

1993 - 1994

CENTRAL AFRICAN REP.

1996 - 1997

CONGO

1998 - 1999

GUINEA BISSAU

1998 - 1999

```r
formattable(wars_country_year_A,  align = c("l", rep("r", NCOL(wars_country_year_A) - 1)))
```

Asia

Conflict Time Frame

CHINA

1946 - 1999

PHILIPPINES

1946 - 1999

BURMA

1948 - 1999

KOREA, S.

1949 - 1950

INDONESIA

1950 - 1999

INDIA

1952 - 1999

LAOS

1960 - 1973

VIETNAM, S.

1960 - 1975

CAMBODIA

1970 - 1992

PAKISTAN

1971 - 1999

SRI LANKA

1971 - 1999

BANGLADESH

1976 - 1997

AFGHANISTAN

1978 - 1999

PAPUA N.G.

1988 - 1998

NEPAL

1997 - 1999

```r
formattable(wars_country_year_NA,  align = c("l", rep("r", NCOL(wars_country_year_NA) - 1)))
```

North Africa and Middle East

Conflict Time Frame

YEMEN ARAB REP.

1948 - 1969

LEBANON

1958 - 1990

IRAQ

1959 - 1974

ALGERIA

1962 - 1999

JORDAN

1970 - 1970

CYPRUS

1974 - 1974

MOROCCO

1975 - 1988

TURKEY

1977 - 1999

IRAN

1978 - 1993

YEMEN PEOP. REP.

1986 - 1987

YEMEN

1994 - 1994

```
repdata <- repdata %>% group_by(war) %>% mutate(outcome = ifelse(war == 1 ,'civil_war','no_war'))
repdata$outcome <- as.factor(repdata$outcome)
# View(repdata)

set.seed(42)
ndx <- sample(nrow(repdata), floor(nrow(repdata) * 0.9))
train <- repdata[ndx,]
test <- repdata[-ndx,]

xTrain <- train[,-70]
yTrain <- train$outcome

xTest <- test[,-70]
yTest <- test$outcome

# model <- naiveBayes(xTrain, yTrain)
# summary(model)
```

```r
model1 <- naiveBayes(outcome ~ warl + lpopl1 + lmtnest
+ ncontig + nwstate + instab + polity2l + ethfrac + relfrac,  data = train, family = "binomial")
summary(model1)
```

```
##           Length Class  Mode
## apriori   2      table  numeric
## tables    9      -none- list
## levels    2      -none- character
## isnumeric 9      -none- logical
## call      5      -none- call
```

```r
df1 <- data.frame(actual = yTest, pred = predict(model1, test))
```

```
## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion
```

```r
head(df1)
```

```
##   actual   pred
## 1 no_war no_war
## 2 no_war no_war
## 3 no_war no_war
## 4 no_war no_war
## 5 no_war no_war
## 6 no_war no_war
```

```r
table(df1)
```

```
##            pred
## actual      civil_war no_war
##   civil_war        89      9
##   no_war            9    554
```

```r
# accuracy: fraction of correct classifications
df1 %>%
  summarize(acc = mean(pred == actual))
```

```
##         acc
## 1 0.9727685
```

```r
# precision: fraction of positive predictions that are actually true
df1 %>%
  filter(pred == 'civil_war') %>%
  summarize(prec = mean(actual == 'civil_war'))
```

```
##        prec
## 1 0.9081633
```

```r
# recall: fraction of true examples that we predicted to be positive
# aka true positive rate, sensitivity
df1 %>%
  filter(actual == 'civil_war') %>%
  summarize(recall = mean(pred == 'civil_war'))
```

```
##       recall
## 1 0.9081633
```

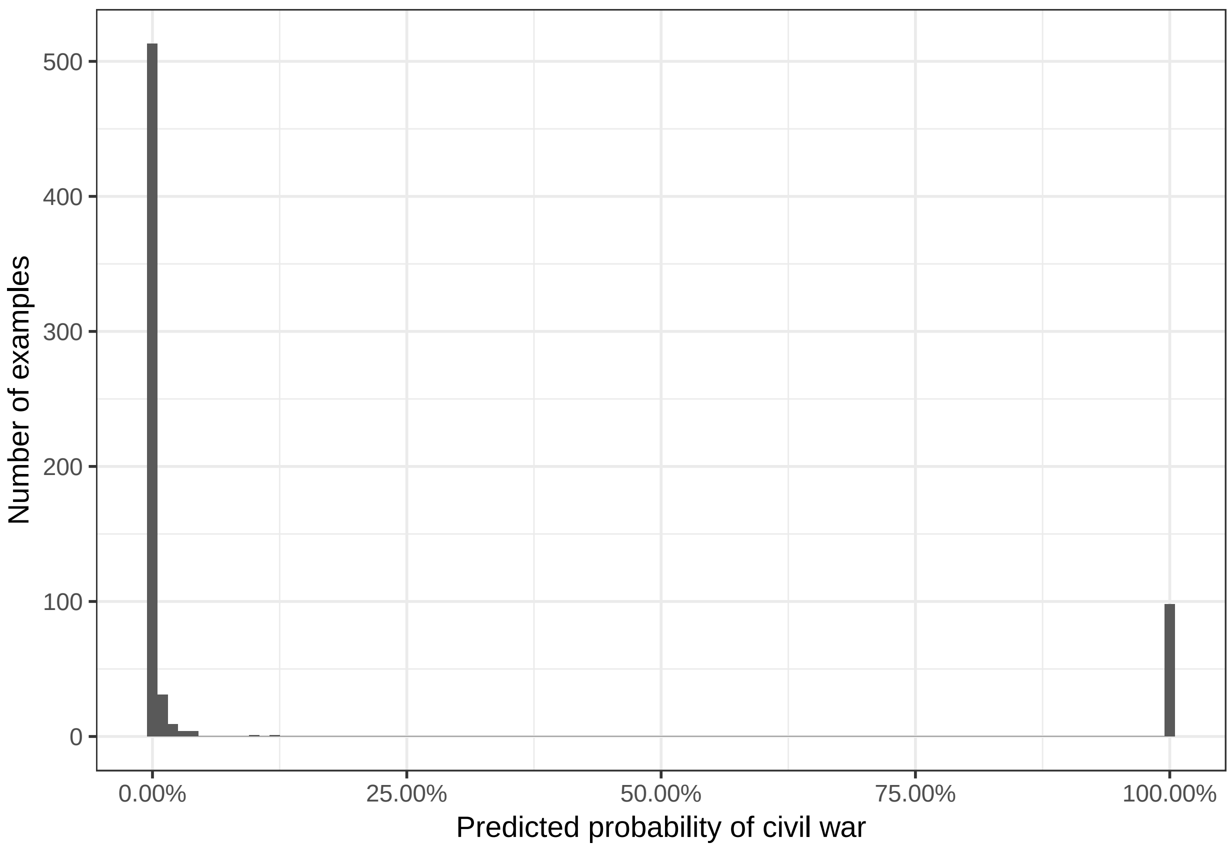
```
# false positive rate: fraction of false examples that we predicted to be positive
df1 %>%
  filter(actual == 'no_war') %>%
  summarize(fpr = mean(pred == 'civil_war'))
```

```
##          fpr
## 1 0.01598579
```

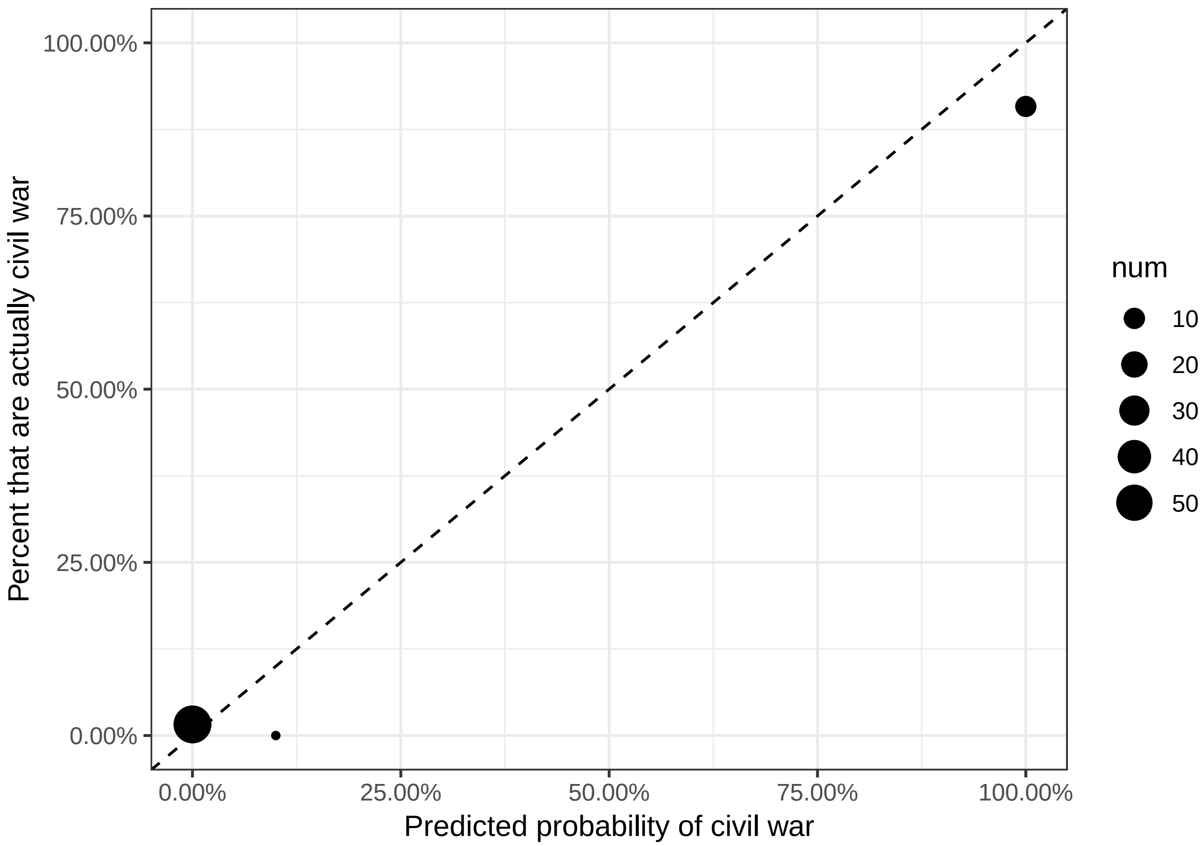
```
# plot histogram of predicted probabilities
# note overconfident predictions
probs1 <- data.frame(predict(model1, test, type="raw"))
```

```
## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion
```

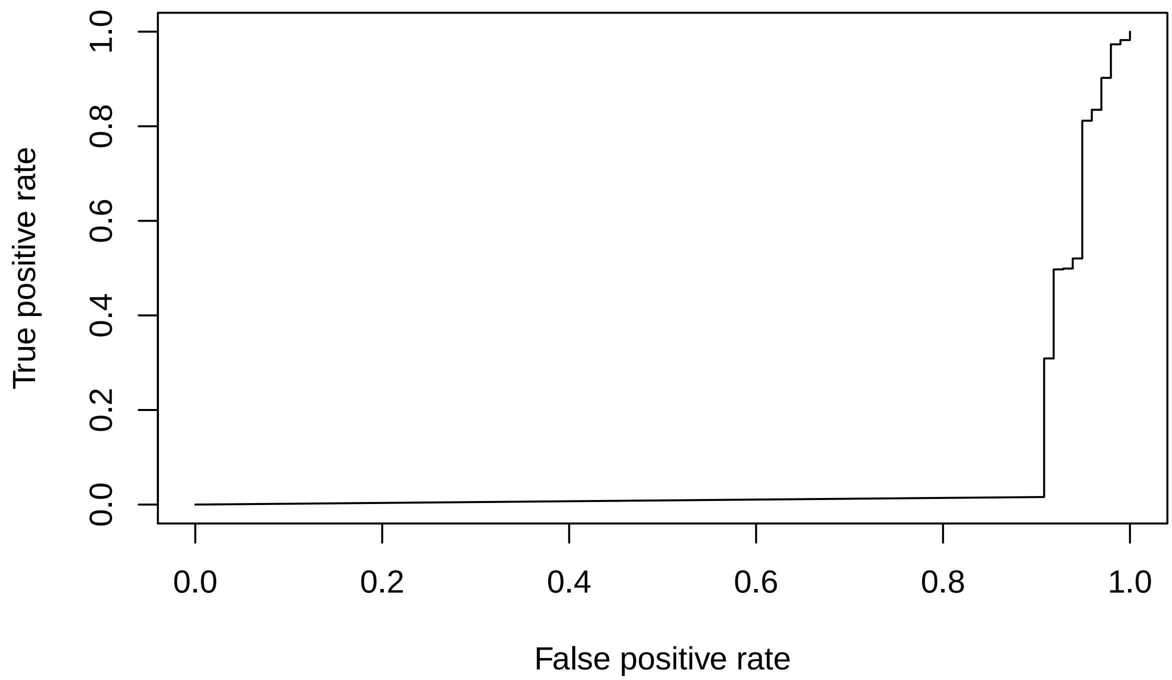
```
ggplot(probs1, aes(x = civil_war)) +
  geom_histogram(binwidth = 0.01) +
  scale_x_continuous(label = percent) +
  xlab('Predicted probability of civil war') +
  ylab('Number of examples')
```



```r
data.frame(predicted=probs1[, "civil_war"], actual=yTest) %>%
  group_by(predicted=round(predicted*10)/10) %>%
  summarize(num=n(), actual=mean(actual == "civil_war")) %>%
  ggplot(data=., aes(x=predicted, y=actual, size=num)) +
  geom_point() +
  geom_abline(linetype=2) +
  scale_x_continuous(labels=percent, lim=c(0,1)) +
  scale_y_continuous(labels=percent, lim=c(0,1)) +
  xlab('Predicted probability of civil war') +
  ylab('Percent that are actually civil war')
```




```r
# create a ROCR object
pred1 <- prediction(probs1[, "civil_war"], yTest)

# create a ROCR object
pred1 <- prediction(probs1[, "civil_war"], yTest)

# plot ROC curve
perf_nb1 <- performance(pred1, measure='tpr', x.measure='fpr')
plot(perf_nb1)
```




```r
performance(pred1, 'auc')
```

```
## An object of class "performance"
## Slot "x.name":
## [1] "None"
##
## Slot "y.name":
## [1] "Area under the ROC curve"
##
## Slot "alpha.name":
## [1] "none"
##
## Slot "x.values":
## list()
##
## Slot "y.values":
## [[1]]
## [1] 0.0718545
##
##
## Slot "alpha.values":
## list()
```

```r
model1 <- naiveBayes(outcome ~ warl + lpopl1 + lmtnest
+ ncontig + nwstate + instab + polity2l + ethfrac + relfrac,  data = train, family = "binomial")
summary(model1)
```

```
##           Length Class  Mode
## apriori   2      table  numeric
## tables    9      -none- list
## levels    2      -none- character
## isnumeric 9      -none- logical
## call      5      -none- call
```

```r
df1 <- data.frame(actual = yTest, pred = predict(model1, test))
```

```
## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion
```

```r
head(df1)
```

```
##   actual   pred
## 1 no_war no_war
## 2 no_war no_war
## 3 no_war no_war
## 4 no_war no_war
## 5 no_war no_war
## 6 no_war no_war
```

```r
table(df1)
```

```
##             pred
## actual      civil_war no_war
##   civil_war        89      9
##   no_war            9    554
```

```r
# accuracy: fraction of correct classifications
df1 %>%
  summarize(acc = mean(pred == actual))
```

```
##         acc
## 1 0.9727685
```

```r
# precision: fraction of positive predictions that are actually true
df1 %>%
  filter(pred == 'civil_war') %>%
  summarize(prec = mean(actual == 'civil_war'))
```

```
##        prec
## 1 0.9081633
```

```r
# recall: fraction of true examples that we predicted to be positive
# aka true positive rate, sensitivity
df1 %>%
  filter(actual == 'civil_war') %>%
  summarize(recall = mean(pred == 'civil_war'))
```

```
##      recall
## 1 0.9081633
```

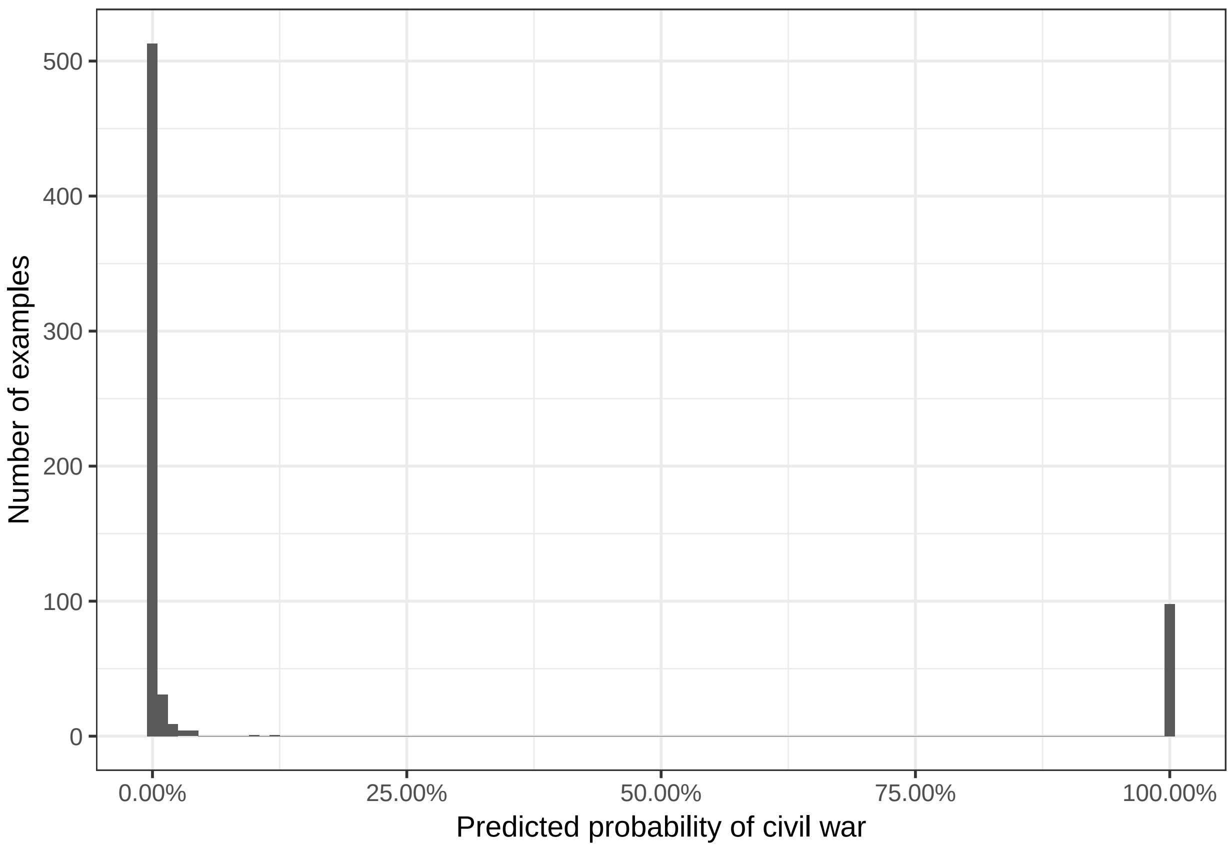
```r
# false positive rate: fraction of false examples that we predicted to be positive
df1 %>%
  filter(actual == 'no_war') %>%
  summarize(fpr = mean(pred == 'civil_war'))
```

```
##          fpr
## 1 0.01598579
```

```r
# plot histogram of predicted probabilities
# note overconfident predictions
probs1 <- data.frame(predict(model1, test, type="raw"))
```

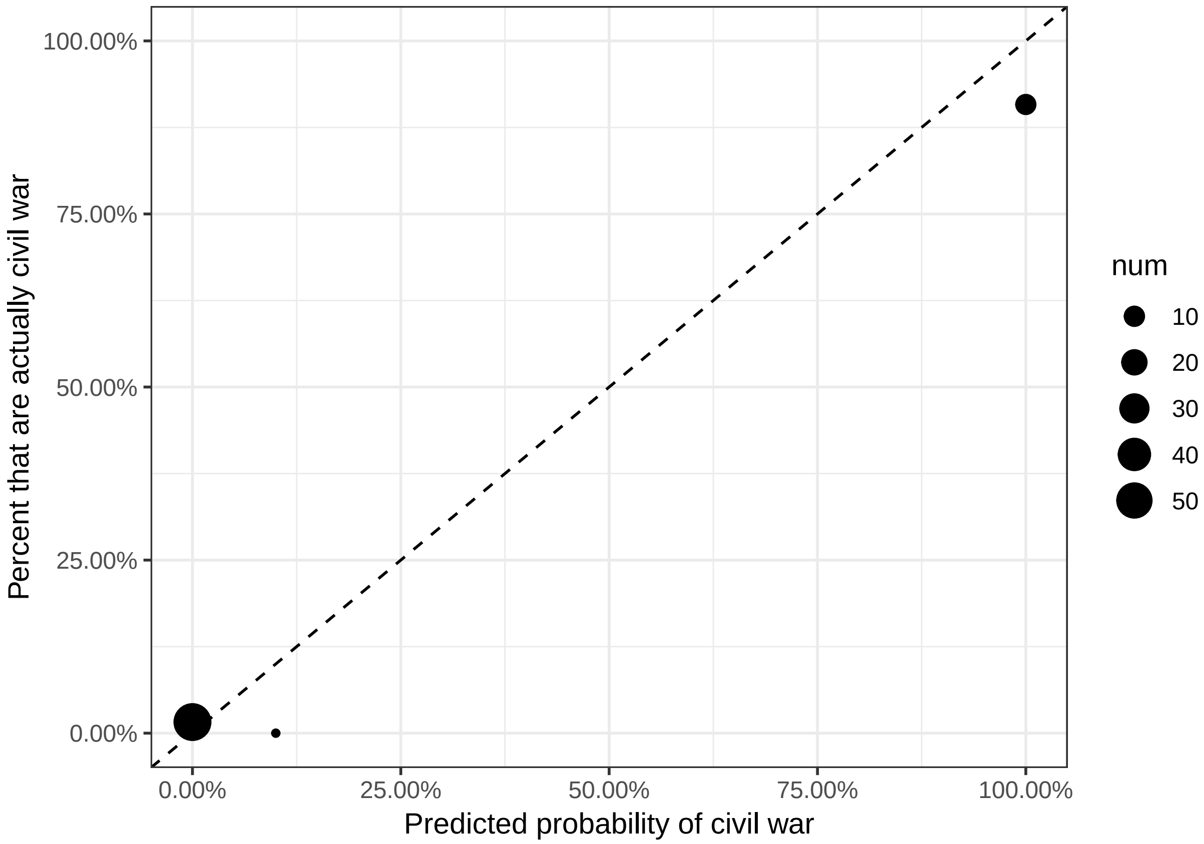
```
## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion
```

```r
ggplot(probs1, aes(x = civil_war)) +
  geom_histogram(binwidth = 0.01) +
  scale_x_continuous(label = percent) +
  xlab('Predicted probability of civil war') +
  ylab('Number of examples')
```

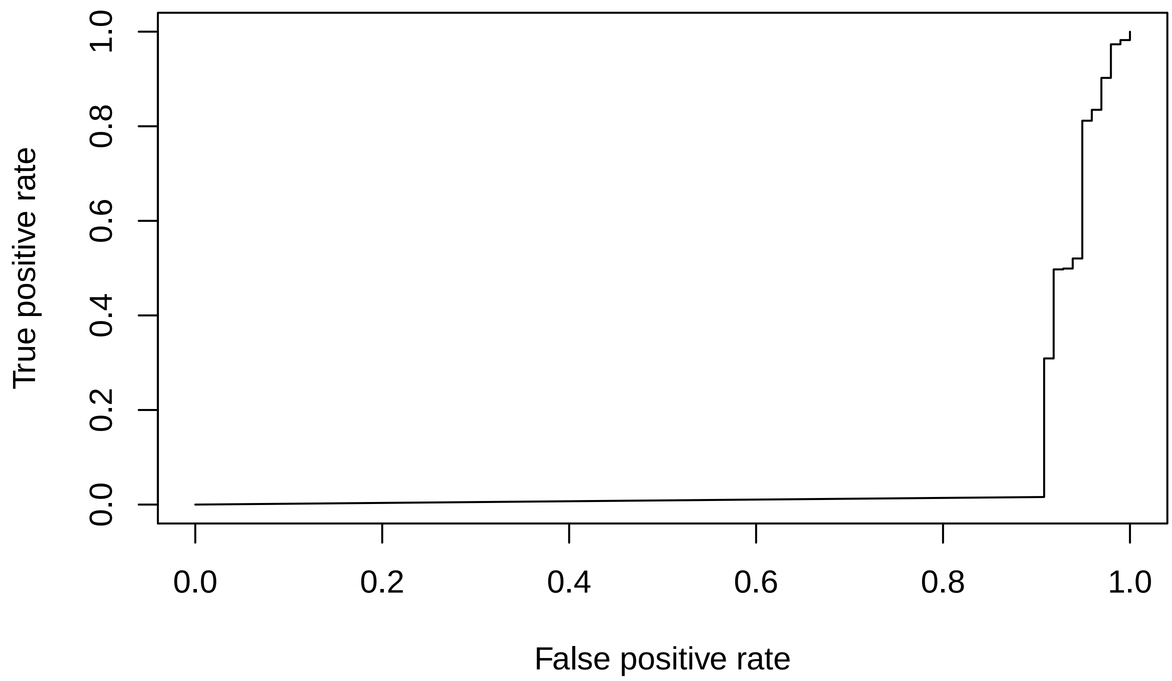



```r
data.frame(predicted=probs1[, "civil_war"], actual=yTest) %>%
  group_by(predicted=round(predicted*10)/10) %>%
  summarize(num=n(), actual=mean(actual == "civil_war")) %>%
  ggplot(data=., aes(x=predicted, y=actual, size=num)) +
  geom_point() +
  geom_abline(linetype=2) +
  scale_x_continuous(labels=percent, lim=c(0,1)) +
  scale_y_continuous(labels=percent, lim=c(0,1)) +
  xlab('Predicted probability of civil war') +
```

```
  ylab('Percent that are actually civil war')
```




```
# create a ROCR object
pred1 <- prediction(probs1[, "civil_war"], yTest)

# create a ROCR object
pred1 <- prediction(probs1[, "civil_war"], yTest)

# plot ROC curve
perf_nb1 <- performance(pred1, measure='tpr', x.measure='fpr')
plot(perf_nb1)
```




```r
performance(pred1, 'auc')
```

```
## An object of class "performance"
## Slot "x.name":
## [1] "None"
##
## Slot "y.name":
## [1] "Area under the ROC curve"
##
## Slot "alpha.name":
## [1] "none"
##
## Slot "x.values":
## list()
##
## Slot "y.values":
## [[1]]
## [1] 0.0718545
##
##
## Slot "alpha.values":
## list()
```

```r
model2 <- naiveBayes(outcome ~ instab ,  data = train, family = "binomial")
summary(model2)
```

```
##           Length Class  Mode
## apriori   2      table  numeric
## tables    1      -none- list
## levels    2      -none- character
## isnumeric 1      -none- logical
## call      5      -none- call
```

```r
df2 <- data.frame(actual = yTest, pred = predict(model2, test))
```

```
## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion
```

```r
head(df2)
```

```
##   actual   pred
## 1 no_war no_war
## 2 no_war no_war
## 3 no_war no_war
## 4 no_war no_war
## 5 no_war no_war
## 6 no_war no_war
```

```r
table(df2)
```

```
##            pred
## actual      civil_war no_war
##   civil_war        25     73
##   no_war           73    490
```

```r
# accuracy: fraction of correct classifications
df2 %>%
  summarize(acc = mean(pred == actual))
```

```
##         acc
## 1 0.7791225
```

```r
# precision: fraction of positive predictions that are actually true
df2 %>%
  filter(pred == 'civil_war') %>%
  summarize(prec = mean(actual == 'civil_war'))
```

```
##       prec
## 1 0.255102
```

```r
# recall: fraction of true examples that we predicted to be positive
# aka true positive rate, sensitivity
df2 %>%
  filter(actual == 'civil_war') %>%
  summarize(recall = mean(pred == 'civil_war'))
```

```
##     recall
## 1 0.255102
```

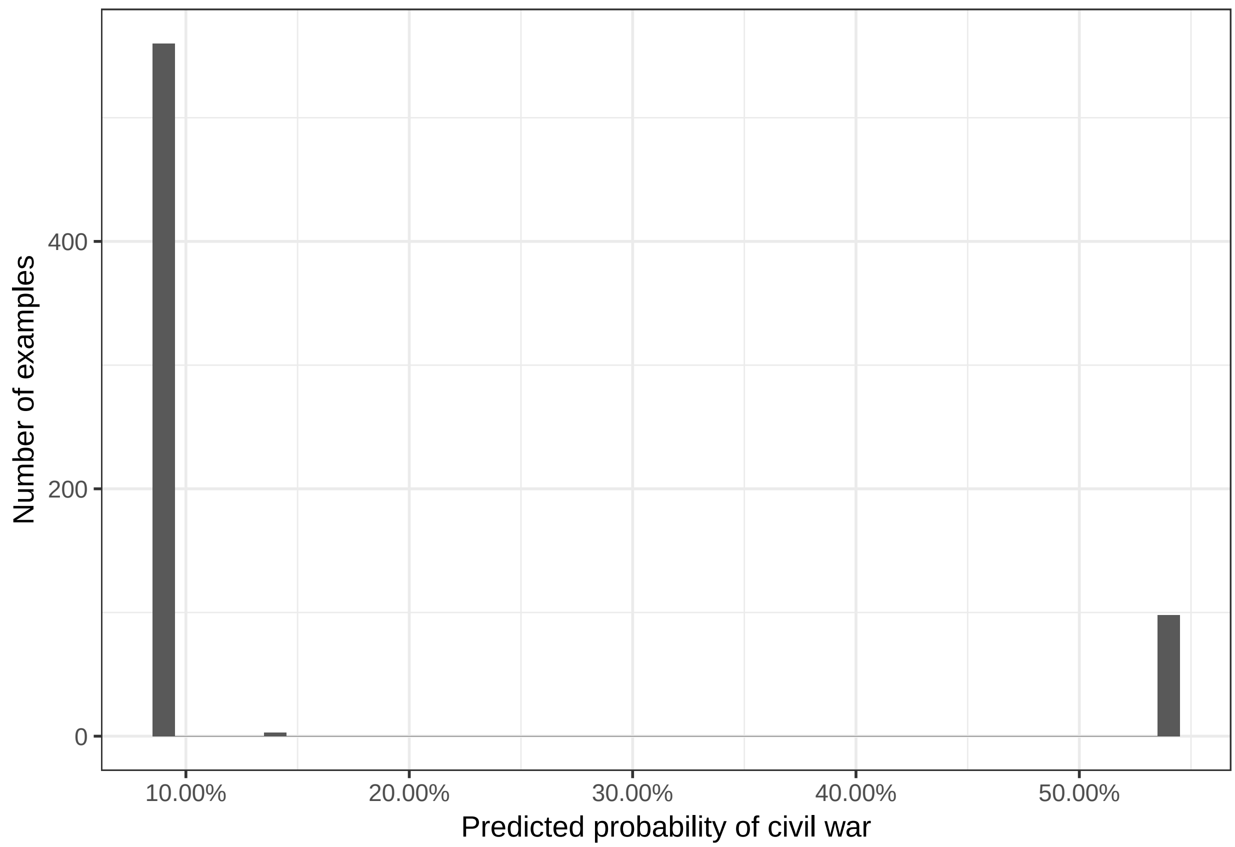
```r
# false positive rate: fraction of false examples that we predicted to be positive
df2 %>%
  filter(actual == 'no_war') %>%
  summarize(fpr = mean(pred == 'civil_war'))
```

```
##         fpr
## 1 0.1296625
```

```r
# plot histogram of predicted probabilities
# note overconfident predictions
probs2 <- data.frame(predict(model2, test, type="raw"))
```

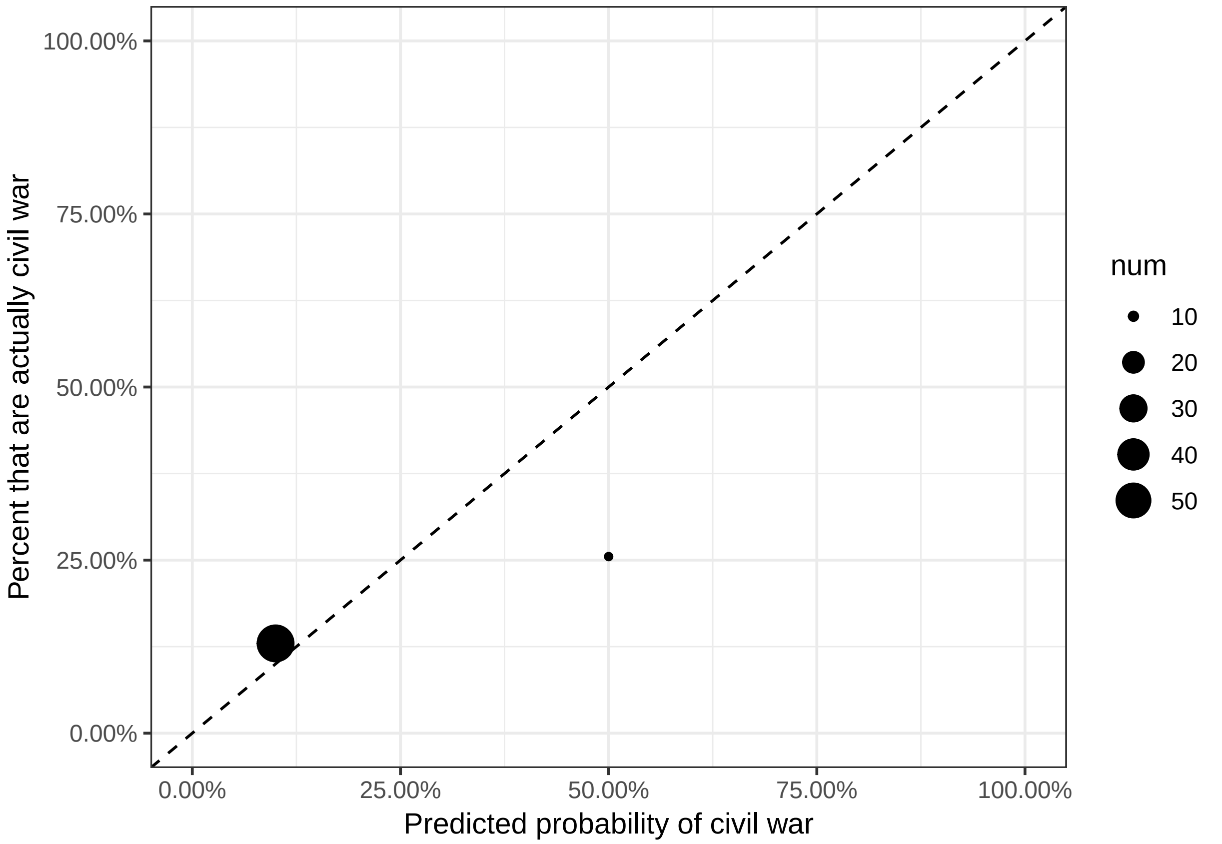
```
## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion

## Warning in data.matrix(newdata): NAs introduced by coercion
```

```r
ggplot(probs2, aes(x = civil_war)) +
  geom_histogram(binwidth = 0.01) +
  scale_x_continuous(label = percent) +
  xlab('Predicted probability of civil war') +
  ylab('Number of examples')
```

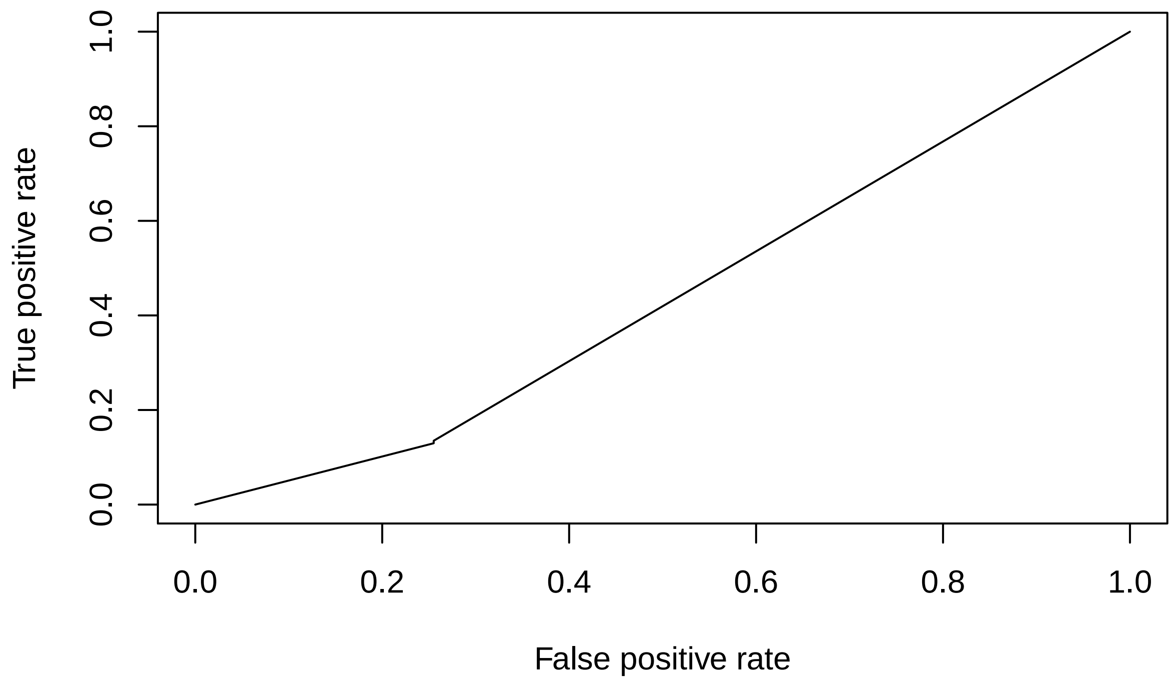



```r
data.frame(predicted=probs2[, "civil_war"], actual=yTest) %>%
  group_by(predicted=round(predicted*10)/10) %>%
  summarize(num=n(), actual=mean(actual == "civil_war")) %>%
  ggplot(data=., aes(x=predicted, y=actual, size=num)) +
  geom_point() +
  geom_abline(linetype=2) +
  scale_x_continuous(labels=percent, lim=c(0,1)) +
  scale_y_continuous(labels=percent, lim=c(0,1)) +
  xlab('Predicted probability of civil war') +
```

```r
  ylab('Percent that are actually civil war')
```




```r
# create a ROCR object
pred2 <- prediction(probs2[, "civil_war"], yTest)

# create a ROCR object
pred2 <- prediction(probs2[, "civil_war"], yTest)

# plot ROC curve
perf_nb2 <- performance(pred2, measure='tpr', x.measure='fpr')
plot(perf_nb2)
```




```r
performance(pred2, 'auc')
```

```
## An object of class "performance"
## Slot "x.name":
## [1] "None"
##
## Slot "y.name":
## [1] "Area under the ROC curve"
##
## Slot "alpha.name":
## [1] "none"
##
## Slot "x.values":
## list()
##
## Slot "y.values":
## [[1]]
## [1] 0.4392649
##
##
## Slot "alpha.values":
## list()
```

```r
sessionInfo()
```

```
## R version 3.5.1 (2018-07-02)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS  10.14.4
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
##
```

```
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] rstanarm_2.18.2     Rcpp_1.0.1          rstan_2.18.2
##  [4] StanHeaders_2.18.1  loo_2.1.0           caret_6.0-84
##  [7] lattice_0.20-35     formattable_0.2.0.1 e1071_1.7-1
## [10] pROC_1.14.0         ROCR_1.0-7          gplots_3.0.1.1
## [13] scales_1.0.0        broom_0.5.2         lubridate_1.7.4
## [16] forcats_0.4.0       stringr_1.4.0       dplyr_0.8.0.1
## [19] purrr_0.3.2         readr_1.3.1         tidyr_0.8.3
## [22] tibble_2.1.1        ggplot2_3.1.1       tidyverse_1.2.1
## [25] foreign_0.8-70
##
## loaded via a namespace (and not attached):
##   [1] minqa_1.2.4        colorspace_1.4-1   class_7.3-14
##   [4] ggridges_0.5.1     rsconnect_0.8.13   markdown_0.9
##   [7] base64enc_0.1-3    rstudioapi_0.10    DT_0.5
##  [10] fansi_0.4.0        prodlim_2018.04.18 xml2_1.2.0
##  [13] codetools_0.2-15   splines_3.5.1      knitr_1.22
##  [16] shinythemes_1.1.2  bayesplot_1.6.0    jsonlite_1.6
##  [19] nloptr_1.2.1       shiny_1.3.2        compiler_3.5.1
##  [22] httr_1.4.0         backports_1.1.4    assertthat_0.2.1
##  [25] Matrix_1.2-14      lazyeval_0.2.2     cli_1.1.0
##  [28] later_0.8.0        htmltools_0.3.6    prettyunits_1.0.2
##  [31] tools_3.5.1        igraph_1.2.4.1     gtable_0.3.0
##  [34] glue_1.3.1         reshape2_1.4.3     cellranger_1.1.0
##  [37] gdata_2.18.0       nlme_3.1-137       crosstalk_1.0.0
##  [40] iterators_1.0.10   timeDate_3043.102  gower_0.2.0
##  [43] xfun_0.6           ps_1.3.0           lme4_1.1-21
##  [46] rvest_0.3.3        miniUI_0.1.1.1     mime_0.6
##  [49] gtools_3.8.1       MASS_7.3-50        zoo_1.8-5
##  [52] ipred_0.9-9        colourpicker_1.0   hms_0.4.2
##  [55] promises_1.0.1     parallel_3.5.1     inline_0.3.15
##  [58] shinystan_2.5.0    yaml_2.2.0         gridExtra_2.3
##  [61] rpart_4.1-13       stringi_1.4.3      dygraphs_1.1.1.6
##  [64] foreach_1.4.4      caTools_1.17.1.2   boot_1.3-20
##  [67] pkgbuild_1.0.3     lava_1.6.5         rlang_0.3.4
##  [70] pkgconfig_2.0.2    matrixStats_0.54.0 bitops_1.0-6
##  [73] evaluate_0.13      labeling_0.3       rstantools_1.5.1
##  [76] recipes_0.1.5      htmlwidgets_1.3    processx_3.3.0
##  [79] tidyselect_0.2.5   plyr_1.8.4         magrittr_1.5
##  [82] R6_2.4.0           generics_0.0.2     pillar_1.3.1
##  [85] haven_2.1.0        withr_2.1.2        xts_0.11-2
##  [88] survival_2.42-3    nnet_7.3-12        modelr_0.1.4
##  [91] crayon_1.3.4       utf8_1.1.4         KernSmooth_2.23-15
##  [94] rmarkdown_1.12     readxl_1.3.1       data.table_1.12.2
##  [97] callr_3.2.0        ModelMetrics_1.2.2 threejs_0.3.1
## [100] digest_0.6.18      xtable_1.8-4       httpuv_1.5.1
## [103] stats4_3.5.1       munsell_0.5.0      shinyjs_1.0
```