

Prediction and cross validation

Soc Stats Reading Group

Alex Kindel

Princeton University

1 December 2016

Outline

- ① Civil war
- ② Cross validation
- ③ Back to civil war
- ④ Why care about prediction?

Ward, Greenhill & Bakke (2010)

- “The perils of policy by p-value: Predicting civil conflicts.” *Journal of Peace Research* 47(4), 363-75.
- “... basing policy prescriptions on statistical summaries of probabilistic models (which are predictions) can lead to misleading policy prescriptions if out-of-sample predictive heuristics are ignored.”
 - ▶ In a word: overfitting

Civil wars

Table I. Variables included in the Fearon & Laitin model

<i>Variable</i>	<i>Statistically significant at 0.05 level</i>
Prior War	Yes
GDP per capita	Yes
Population	Yes
Mountainous Terrain	Yes
Non-contiguous State	No
Oil Exporter	Yes
New State	Yes
Instability	Yes
Democracy	No
Ethnic Fractionalization	No
Religious Fractionalization	No

* based on Fearon and Laitin, 2003: Table 1, Column 1.

Table II. Variables included in the Collier & Hoeffler model

<i>Variable</i>	<i>Statistically significant at 0.05 level</i>
Commodity Dependence	Yes
Squared Commodity Dependence	Yes
Male Secondary Schooling	Yes
GDP Growth	Yes
Peace Duration	Yes
Geographic Dispersion	Yes
Population	Yes
Social Fractionalization	Yes
Ethnic Dominance	No

*based on Collier and Hoeffler, 2004: Table 5, column 5.

- Based on logistic regression
- Widely used to guide policy
 - ▶ World Bank, House of Representatives
 - ▶ *The New Yorker*, *The New York Times*, etc.

Civil wars

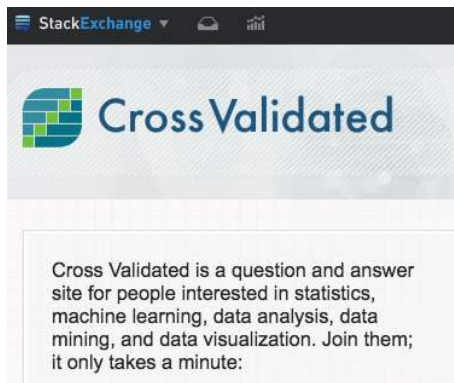
- But: Strikingly poor performance on in-sample prediction

Table III. Number of correctly predicted onsets and false positives at varying cut-points

<i>Fearon & Laitin model</i>		
<i>Threshold</i>	<i>Correctly predicted</i>	<i>False positives</i>
0.5	0/107	0
0.3	1/107	3
0.1	15/107	66

<i>Collier & Hoeffler model</i>		
<i>Threshold</i>	<i>Correctly predicted</i>	<i>False positives</i>
0.5	3/46	5
0.3	10/46	20
0.1	34/46	110

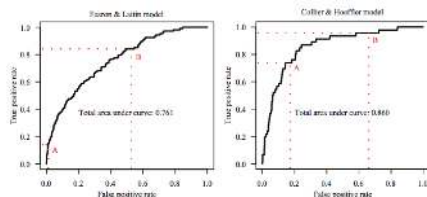
Cross validation



Procedure

- ① Split data into k “folds” (equally sized groups)
- ② Withholding one fold, re-estimate model
- ③ Test predictive power of model on withheld group (AUC)

Receiver operating characteristic (ROC) curve



- We use area under the ROC curve (AUC) as a heuristic measure of predictiveness
 - ▶ Intuitively, increasing AUC implies $TPR > FPR$
- (From the people who brought you instructional television...)

Tricks and missteps

- Bias-variance tradeoff
 - ▶ $k = n$ (LOOCV): higher variance (low variance among training sets), but lower bias
 - ▶ $k < n$ (k -fold): lower variance, but higher bias (*overestimating* prediction error)
- General consensus is that it might be better to overestimate prediction error (conservative bias)
 - ▶ Also, LOOCV is “more expensive”
- **Don't do (supervised) feature selection before model validation!**
 - ▶ Will overestimate AUC (drastically)

Cross validation: pretty easy to implement!

```
# Function to divide data into folds randomly
fold <- function(data, k) {
  data <- data[sample(nrow(data)),] # Shuffle data
  data %<>% mutate(fold = cut(seq(1:nrow(data)), breaks = k, labels=FALSE))
  return(data)
}

# Function to cross-validate data on given model (curried)
cv.predict.logit <- function(data, dv, model.fx, k) {
  data %<>% fold(k) # Fold data
  aucs <- c()
  for(i in 1:k) {
    # Divide data into train and test sets
    train <- data %>% filter(fold != i)
    test <- data %>% filter(fold == i)

    # Estimate model on training data
    mx <- model.fx(data=train)

    # Predict on test data and calculate AUC
    preds <- predict(mx, newdata=test, type="response")
    AUC <- somers2(preds, test[[dv]])[1]
    aucs[i] <- AUC
  }
  return(mean(aucs, na.rm=TRUE)) # Yield mean AUC
}

# Function to rerun CV results n times and average AUCs
crossval <- function(data, dv, model.fx, k, n) {
  aucs <- replicate(n, cv.predict.logit(data, dv, model.fx, k))
  return(aucs)
}
```

Back to civil war

```
# Define Collier & Hoeffler model
```

```
ch.form <- as.factor(warsa) ~ sxp + sxp2 + secsm + gy1 + peace + geo  
ch.mx <- Curry(glm, formula=ch.form, family=binomial(link=logit))
```

```
# Define Fearon & Laitin model
```

```
fl.form <- as.factor(onset) ~ war1 + gdpen1 + lpopl1 + lmtnest + nc  
fl.mx <- Curry(glm, formula=fl.form, family=binomial(link=logit))
```

```
# Perform cross-validation
```

```
k <- 4 # Set k folds
```

```
ch.auc <- cv.predict.logit(ch, "warsa", ch.mx, k)
```

```
fl.auc <- cv.predict.logit(fl, "onset", fl.mx, k)
```

```
c(ch.auc, fl.auc)
```

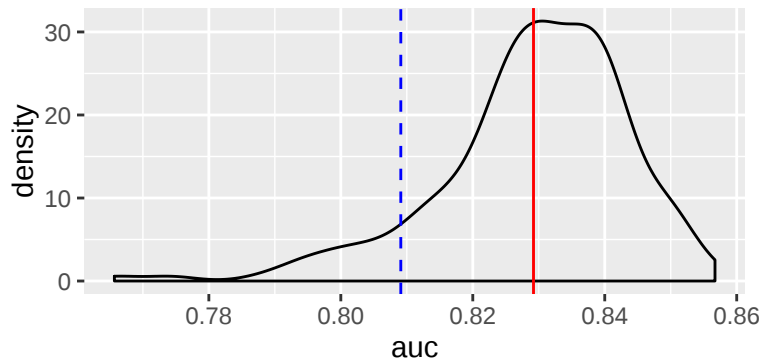
```
## [1] 0.8090876 0.7423249
```

Calculating a stable AUC

- Sensitive to dataset randomization during “folding”
 - ▶ Not too much to worry about here (usually)
- Sensitive to choice of k
 - ▶ Low k : upward bias in AUC
 - ▶ High k : higher variance in AUC

Sensitivity to randomization: F&L

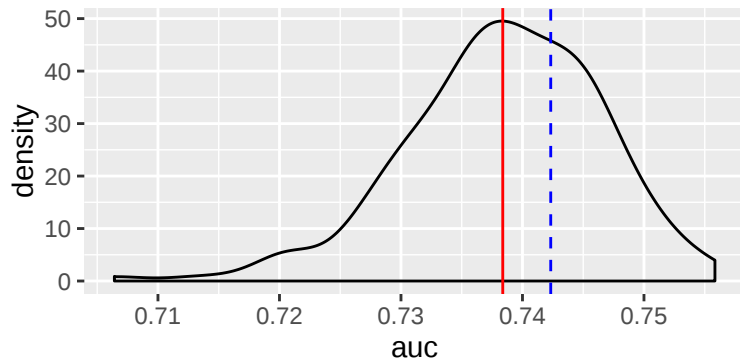
```
k <- 4  
n <- 200 # Set n CV cycles  
ch.aucs <- crossval(ch, "warsa", ch.mx, k, n)
```



- mean over N cycles
- AUC in first cycle

Sensitivity to randomization: C&H

```
k <- 4  
n <- 200 # Set n CV cycles  
fl.aucs <- crossval(fl, "onset", fl.mx, k, n)
```



- mean over N cycles
- AUC in first cycle

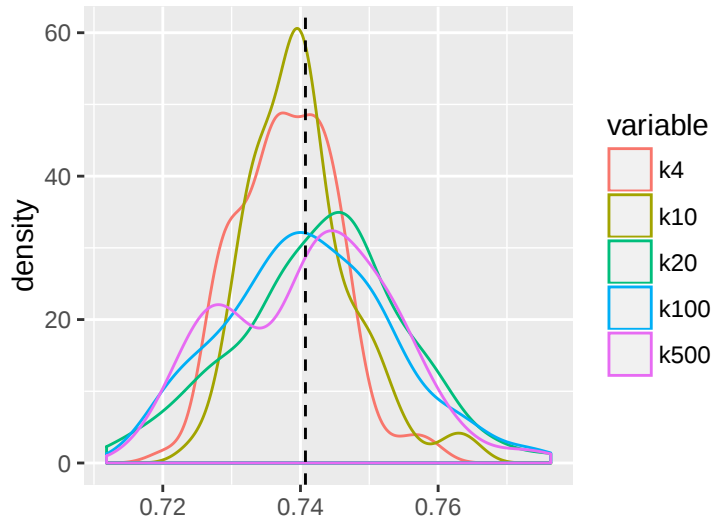
Sensitivity to choice of k : F&L

```
n <- 100
list(k4 = crossval(fl, "onset", fl.mx, 4, n),
     k10 = crossval(fl, "onset", fl.mx, 10, n),
     k20 = crossval(fl, "onset", fl.mx, 20, n),
     k100 = crossval(fl, "onset", fl.mx, 100, n),
     k500 = crossval(fl, "onset", fl.mx, 500, n)) ->
fl.aucs.ks
```

Sensitivity to choice of k : F&L

Using as id variables

Using as id variables



Conclusion: why might we care?

- Technical tradeoff between variable significance vs. model predictiveness (Ward et al. 2010; Lo et al. 2015)
- If we really think our models explain causal effects, shouldn't they be predictive? (Watts 2014)
 - ▶ Especially if we're basing policy on our findings
- Distinguishing origins from effects (Sewell 1996; Pierson 2000; Clemens 2007)