

Advanced Network Analysis

Model Comparison Using ROC Curves

Olga Chyzh [www.olgachyzh.com]

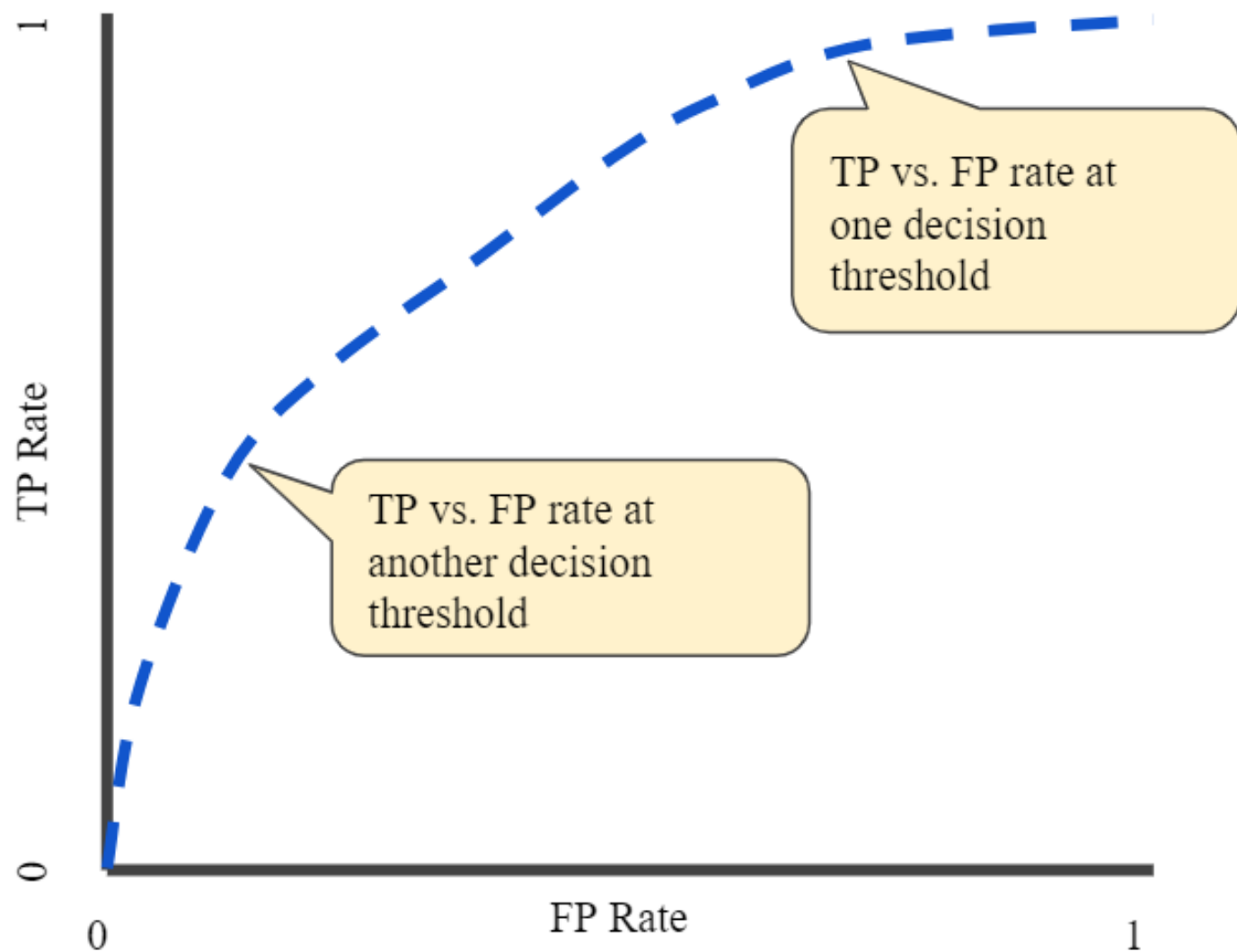
ROC Curves

- An ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:
 - True Positive Rate (recall or sensitivity):

$$TPR = \frac{TP}{TP + FN} = \frac{\textit{correct ones}}{\textit{total ones}}$$

- False Positive Rate (specificity):

$$FPR = \frac{FP}{FP + TN} = \frac{\textit{incorrect ones}}{\textit{total zeroes}}$$



What We Need (Binary DV)

- Our DV (in vector form)
- The predicted probability for each observation based on our model

Example: ERGM

```
library(tidyverse)
library(magrittr)
library(sna)
library(ergm)
data(sampson)

m1 = ergm(samplike ~ edges + nodematch('group'))
summary(m1)
```

```
## Call:
## ergm(formula = samplike ~ edges + nodematch("group"))
##
## Maximum Likelihood Results:
##
##           Estimate Std. Error MCMC % z value Pr(>|z|)
## edges          -2.0015      0.2131      0  -9.393  <1e-04 ***
## nodematch.group   2.6481      0.3026      0   8.751  <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##           Null Deviance: 424.2  on 306  degrees of freedom
##           Residual Deviance: 276.9  on 304  degrees of freedom
##
```

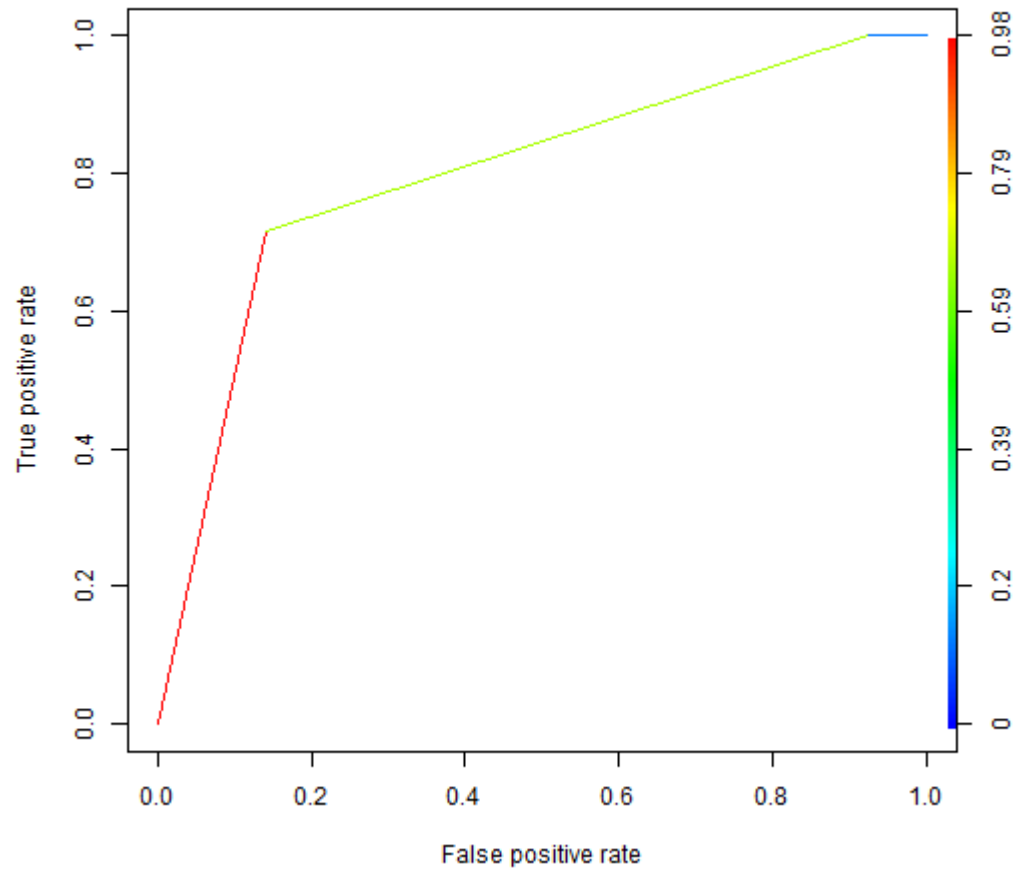
Get Y and Predicted Probabilities

```
myDV<-as.matrix(samplike) %>% data.frame() %>% mutate(name1=row.names  
  pivot_longer(cols=-name1, names_to="name2", values_to="Y")  
  
pred_ergm %>% data.frame() %>% mutate(name1=row.names(.)) %>%  
  pivot_longer(cols=-name1, names_to="name2", values_to="p") %>% left
```

Make a Plot

```
#install.packages("ROCR")  
library(ROCR)  
pred <- prediction(mydat$p, mydat$Y)  
perf <- performance(pred, "tpr", "fpr")  
ergmplot<-plot(perf,colorize=TRUE)
```

Make a Plot



Your Turn

Change the model specification (add an endogenous variable), estimate the new model, and re-make the plot.

Your Turn

1. Transform the data as necessary to estimate an AME model that matches m_1 in specification.
2. Estimate the AME model equivalent to m_1 .

Let's Make an ROC Plot

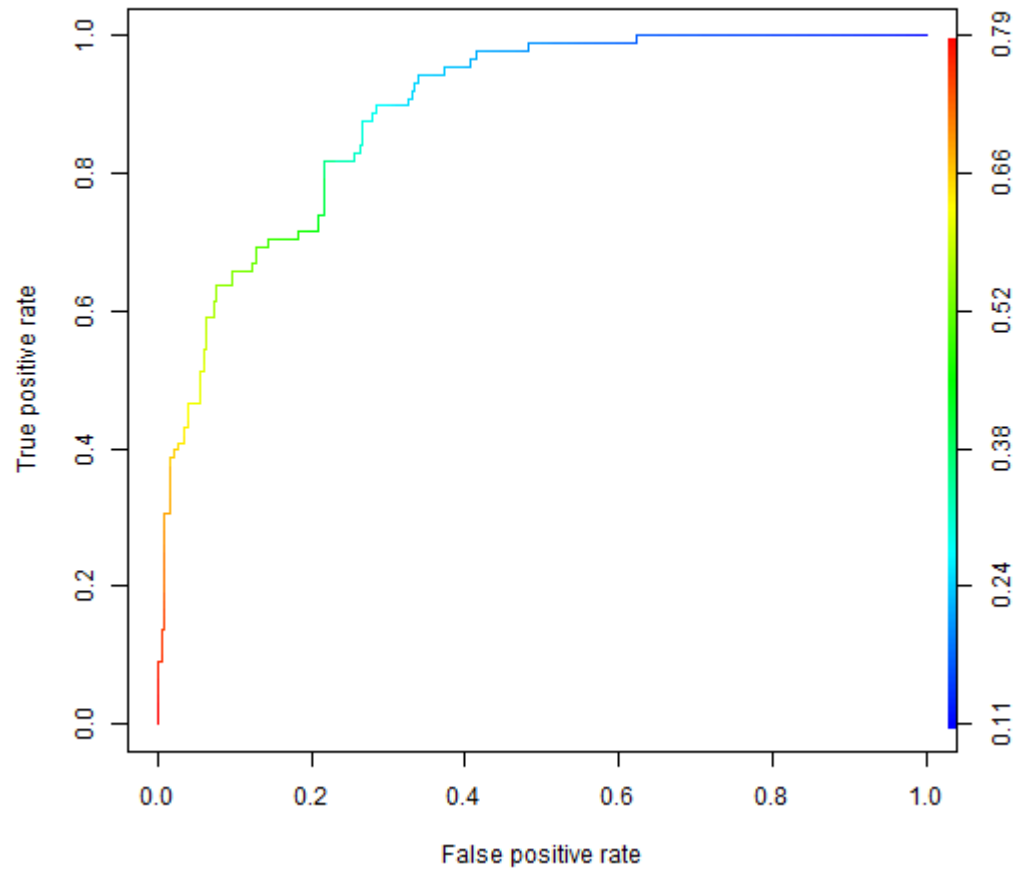
```
lfmFit<-readRDS("data/lfmFit_sampson.rds")

#Get predicted probabilities from AME:
theta<-lfmFit$'EZ' #logged odds ratios for each obs
plogis(as.matrix(theta)) %>% as.data.frame() %>% #convert to probabilities
  mutate(name1=row.names(.)) %>%
  pivot_longer(cols=c(1:18),names_to="name2", values_to="p")->p

as.matrix(samplike) %>% as.data.frame() %>% mutate(name1=row.names(.))
  pivot_longer(cols=-name1, names_to="name2", values_to="Y") %>% left_join(p,by="name2")

pred1 <- prediction(mydat1$p, mydat1$Y)
perf1 <- performance(pred1,"tpr","fpr")
plot(perf1,colorize=TRUE, smooth=FALSE)
```

Let's Make an ROC Plot

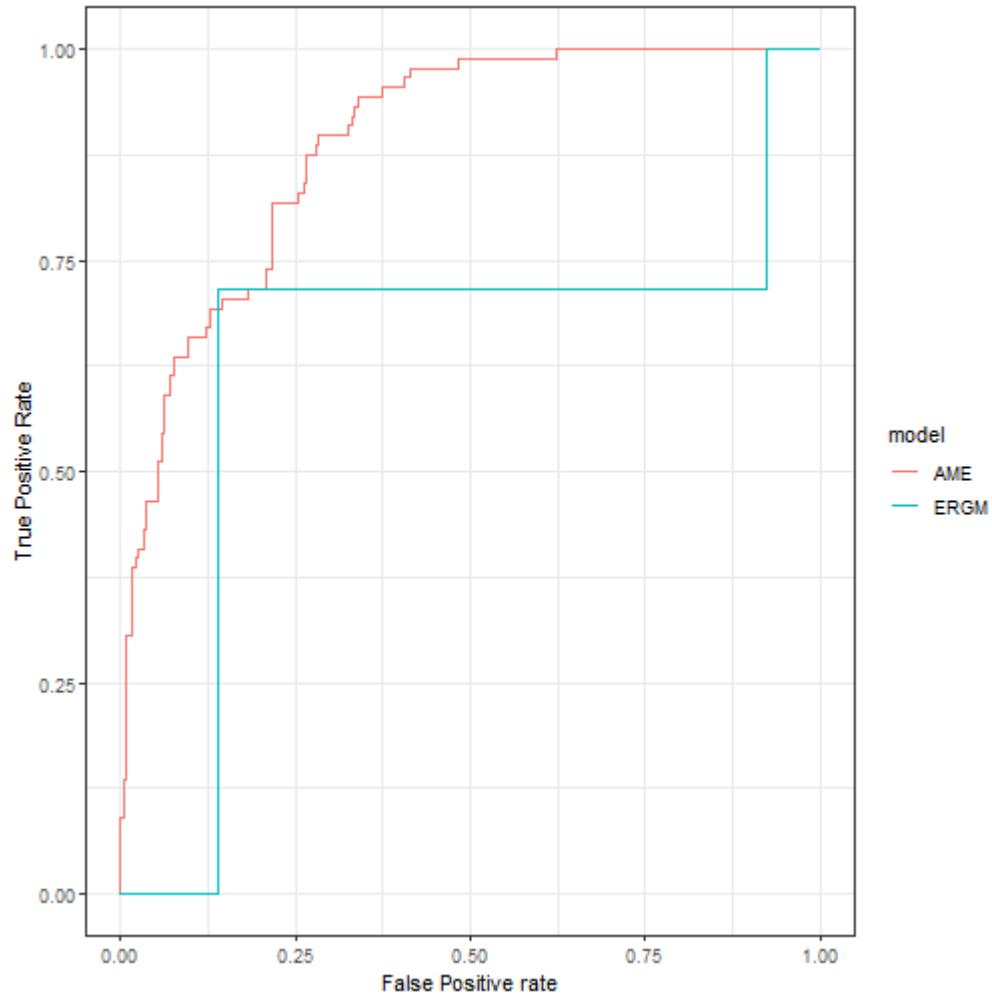


Put These Together

```
library(ggplot2)

ggdata<-rbind.data.frame(cbind.data.frame(x=unlist(perf@x.values),y=unlist(perf@y.values)),
                        cbind.data.frame(x=unlist(perf1@x.values),y=unlist(perf1@y.values)))
ggplot(data=ggdata)+geom_step(aes(x=x, y=y, color=model)) +
  ylab("True Positive Rate")+
  xlab("False Positive rate")+
  theme_bw()
```

Put These Together



Your Turn

1. Re-specify the ERGM to increase the classification power. Plot the ROCs for the two models again.