Finding the right cut-off: the strategy curve

CREDIT RISK MODELING IN R

Lore Dirick

Manager of Data Science Curriculum at Flatiron School

Constructing a confusion matrix

```
predict(log_reg_model, newdata = test_set, type = "response")
```

```
 1          2          3          4          5          ...
0.08825517 0.3502768 0.28632298 0.1657199 0.11264550    ...
```

```
predict(class_tree, new data = test_set)
```

```
          0         1
1 0.7873134 0.2126866
2 0.6250000 0.3750000
3 0.6250000 0.3750000
4 0.7873134 0.2126866
5 0.5756867 0.4243133
```

Cut-off?

```
pred_log_regression_model <- predict(log_reg_model,
                                     newdata = test_set,
                                     type = "response")

cutoff <- 0.14

class_pred_logit <- ifelse(pred_log_regression_model > cutoff, 1, 0)
```

A certain strategy

```
log_model_full <- glm(loan_status ~ ., family = "binomial", data = training_set)

predictions_all_full <- predict(log_reg_model, newdata = test_set, type = "response")

cutoff <- quantile(predictions_all_full, 0.8)
cutoff
```

```
      80%
0.1600124
```

```
pred_full_20 <- ifelse(predictions_all_full > cutoff, 1, 0)
```

```
true_and_predval <- cbind(test_set$loan_status, pred_full_20)
true_and_predval
```

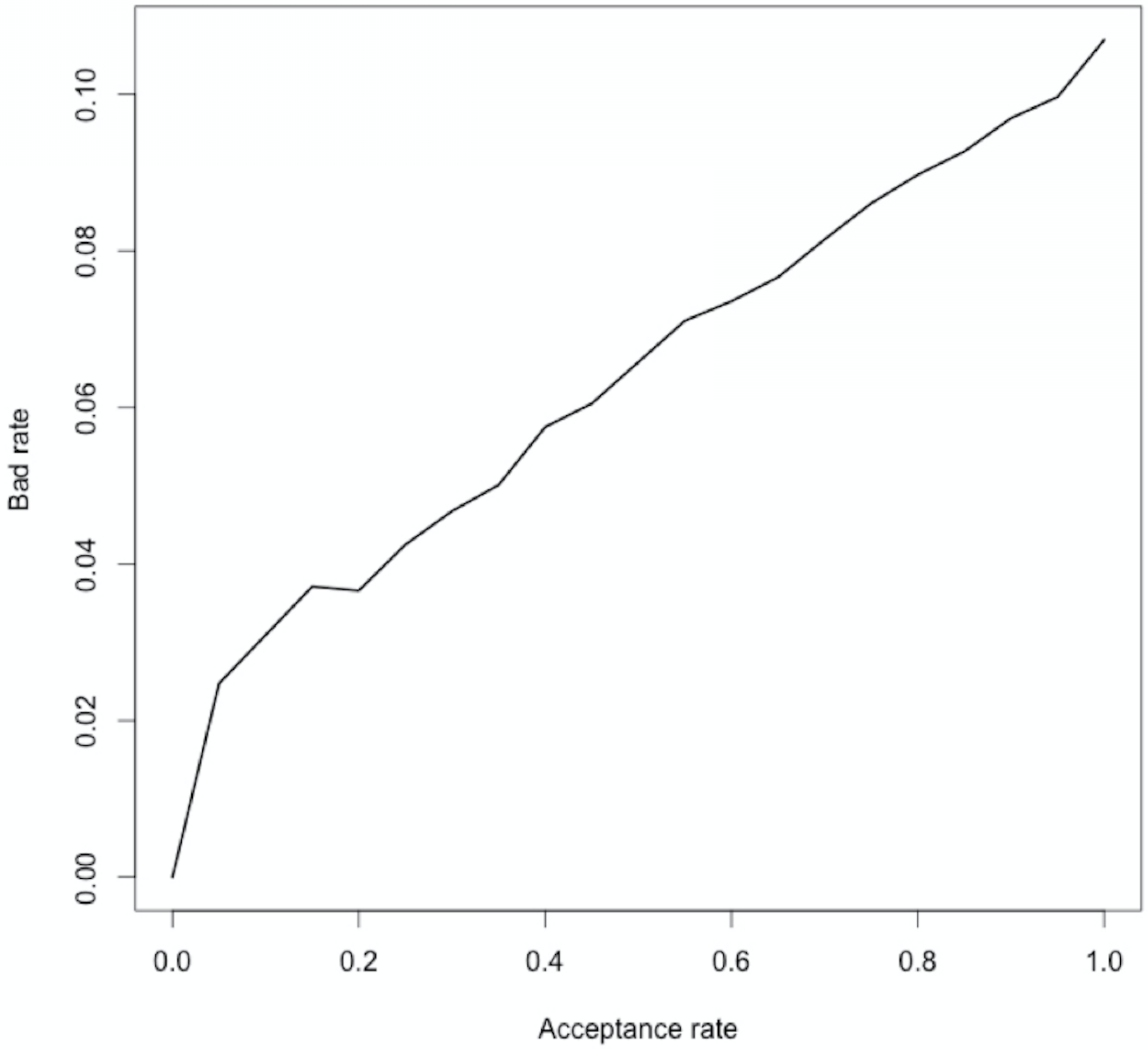
```
     test_set$loan_status    pred_full_20
1          0                      0
2          0                      0
3          0                      1
4          0                      0
5          0                      1
...           ...                      ...
```

```
accepted_loans <- true_and_predval[pred_full_20 == 0,1]
bad_rate <- sum(accepted_loans)/length(accepted_loans)
bad_rate
```

```
0.08972541
```

```
accept_rate  cutoff  bad_rate
 [1,]       1.00     0.5142   0.1069
 [2,]       0.95     0.2122   0.0997
 [3,]       0.90     0.1890   0.0969
 [4,]       0.85     0.1714   0.0927
 [5,]       0.80     0.1600   0.0897
 [6,]       0.75     0.1471   0.0861
 [7,]       0.70     0.1362   0.0815
 [8,]       0.65     0.1268   0.0766
...          ...         ...      ...
[16,]       0.25     0.0644   0.0425
[17,]       0.20     0.0590   0.0366
[18,]       0.15     0.0551   0.0371
[19,]       0.10     0.0512   0.0309
[20,]       0.05     0.0453   0.0247
[21,]       0.00     0.0000   0.0000
```

The strategy curve



Let's practice!

CREDIT RISK MODELING IN R

The ROC-curve

CREDIT RISK MODELING IN R

Lore Dirick

Manager of Data Science Curriculum at Flatiron School

Until now

- Strategy table/curve : still make assumption
- What is "overall" best model?

Confusion matrix

Actual loan status v. Model prediction

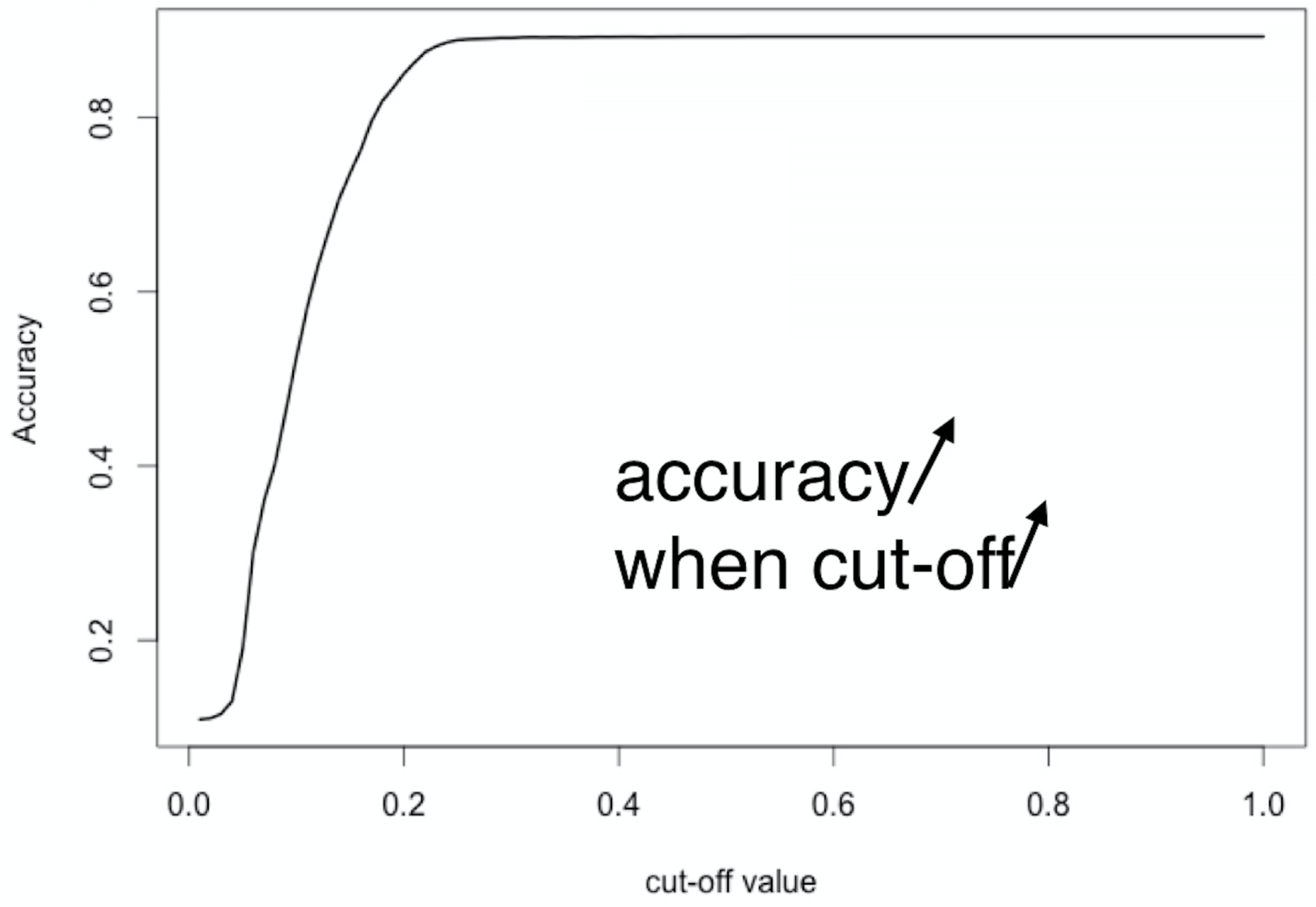
	No default (0)	Default (1)
No default (0)	TN	FP
Default (1)	FN	TP

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN}$$

$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

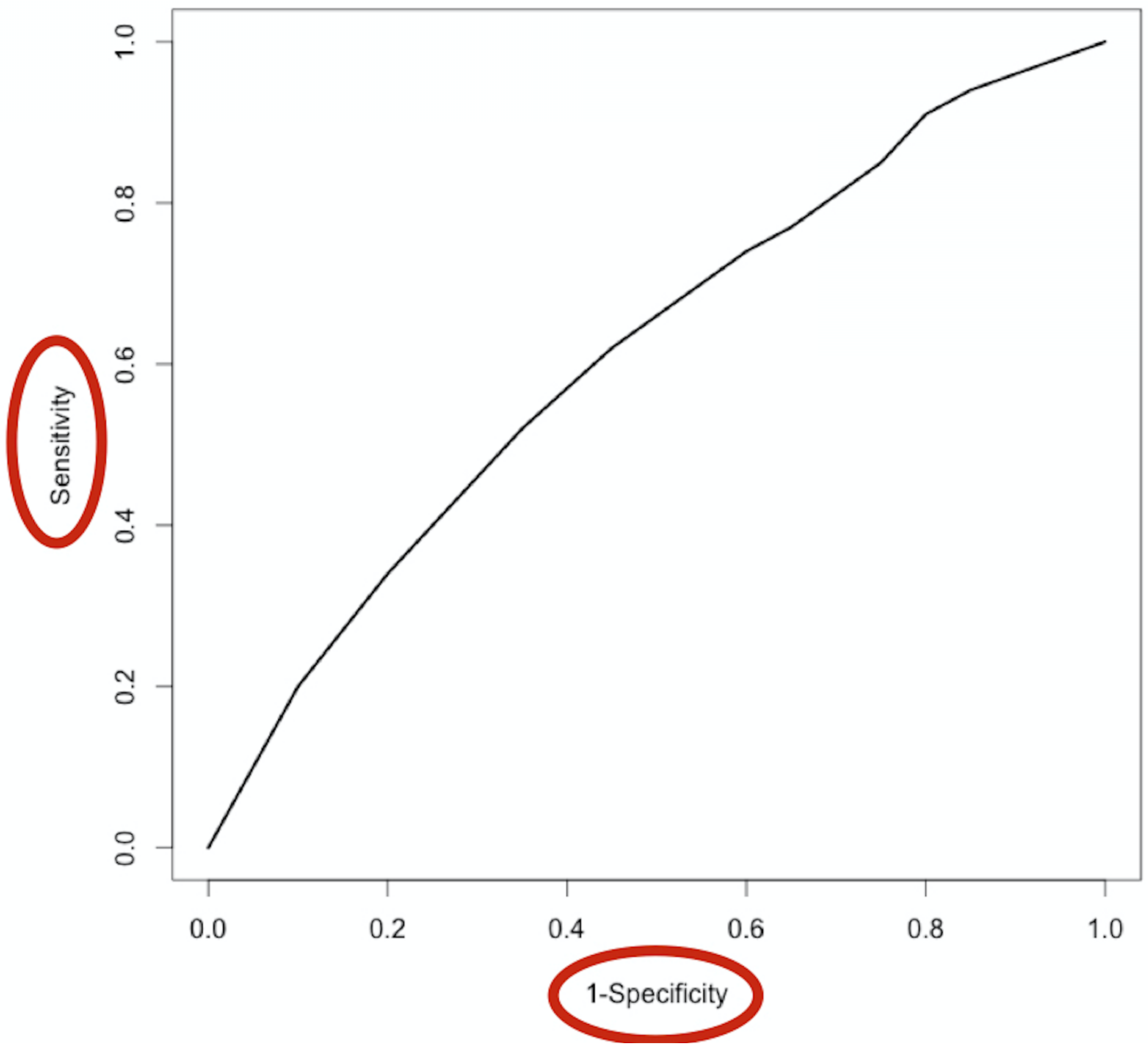
Accuracy?



$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

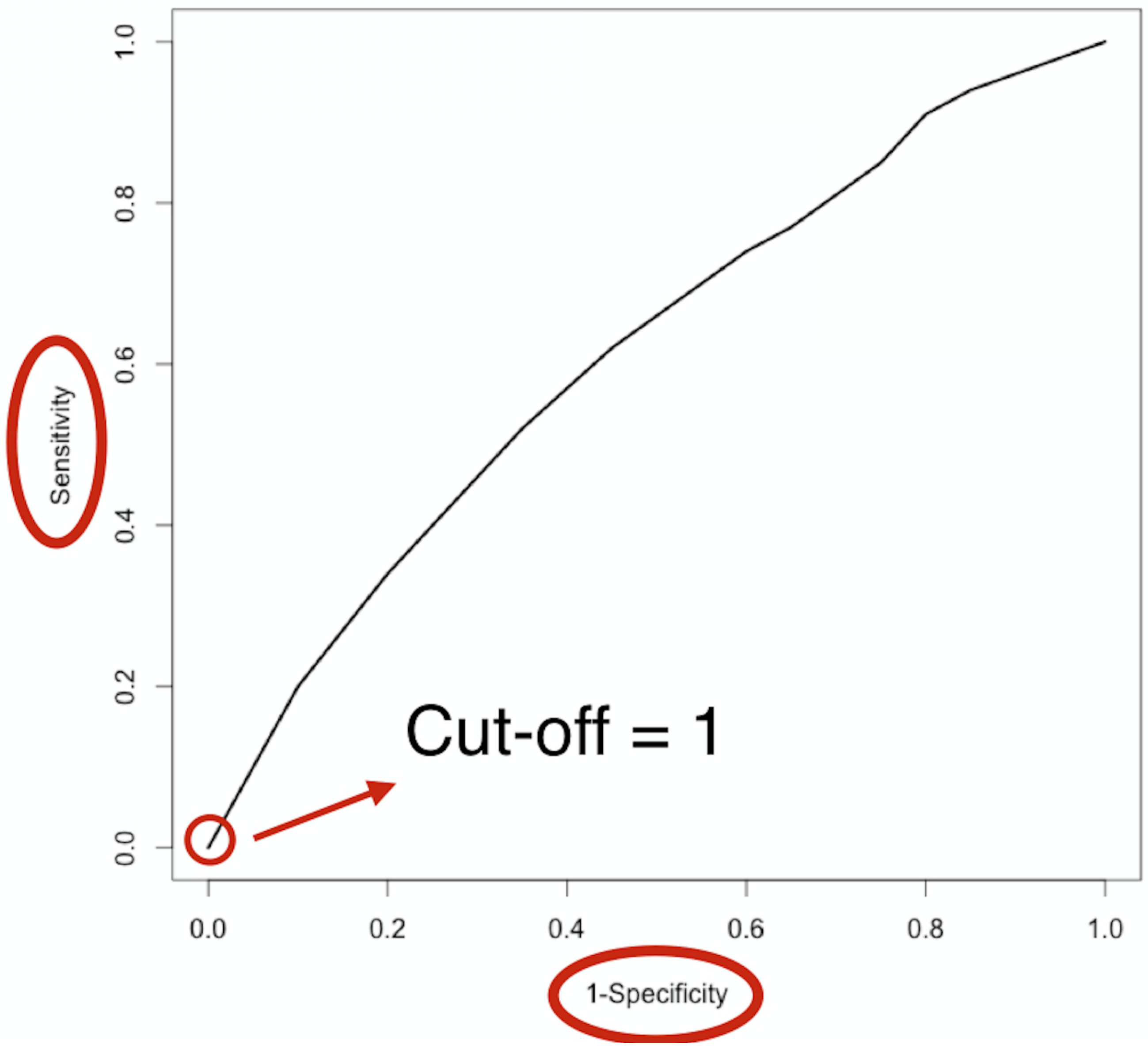
The ROC-curve



$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

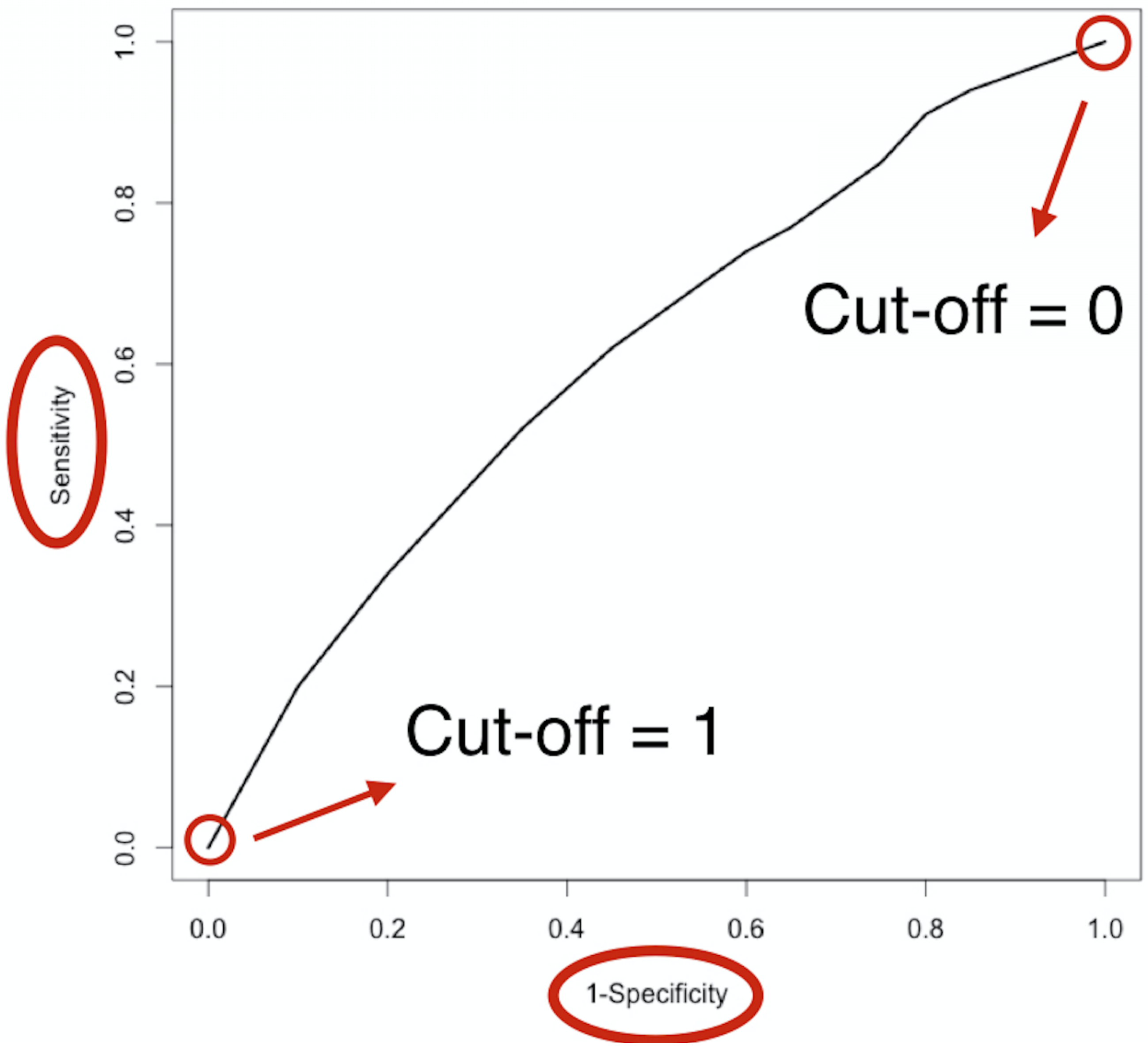
The ROC-curve



$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

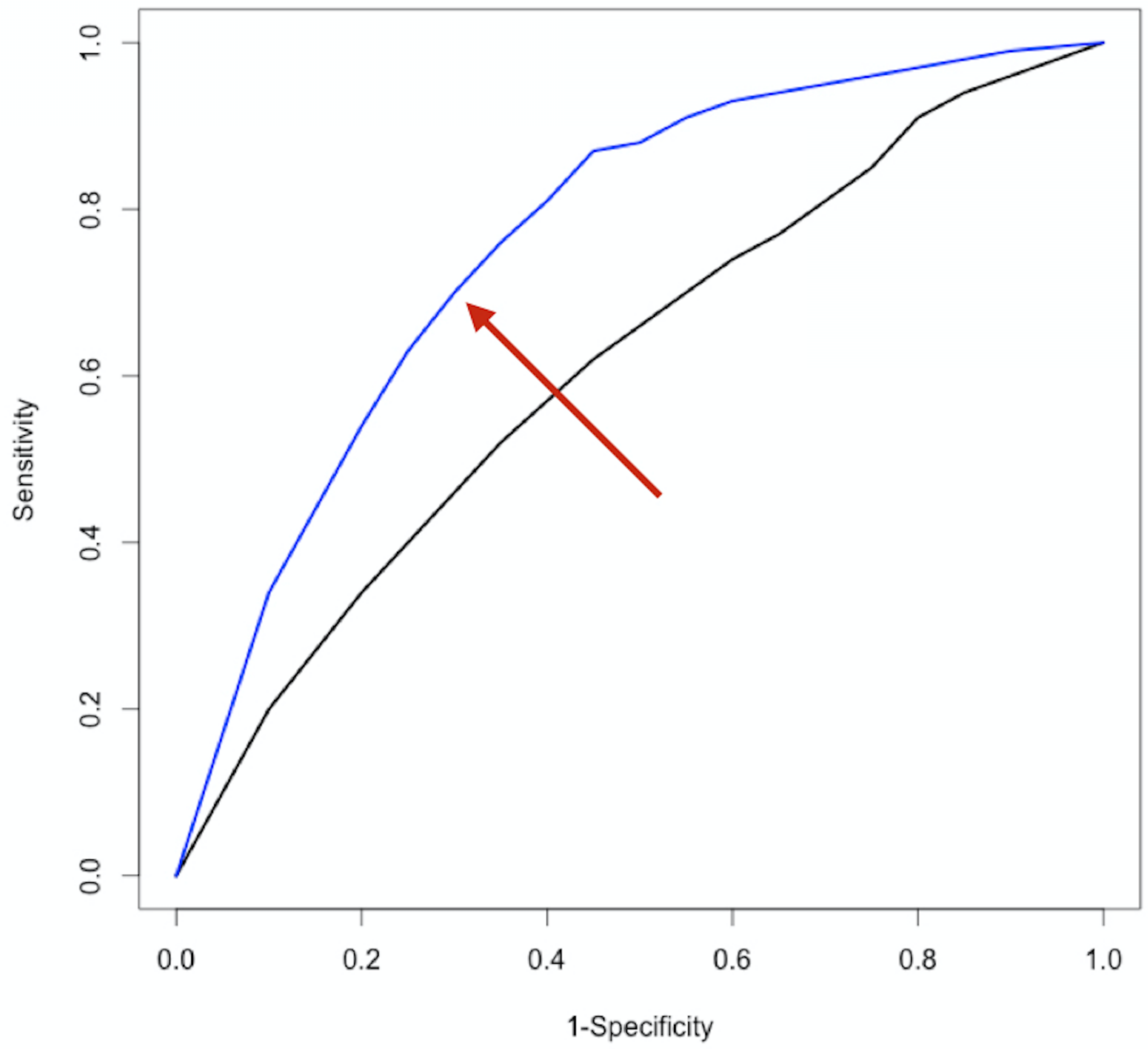
The ROC-curve



$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

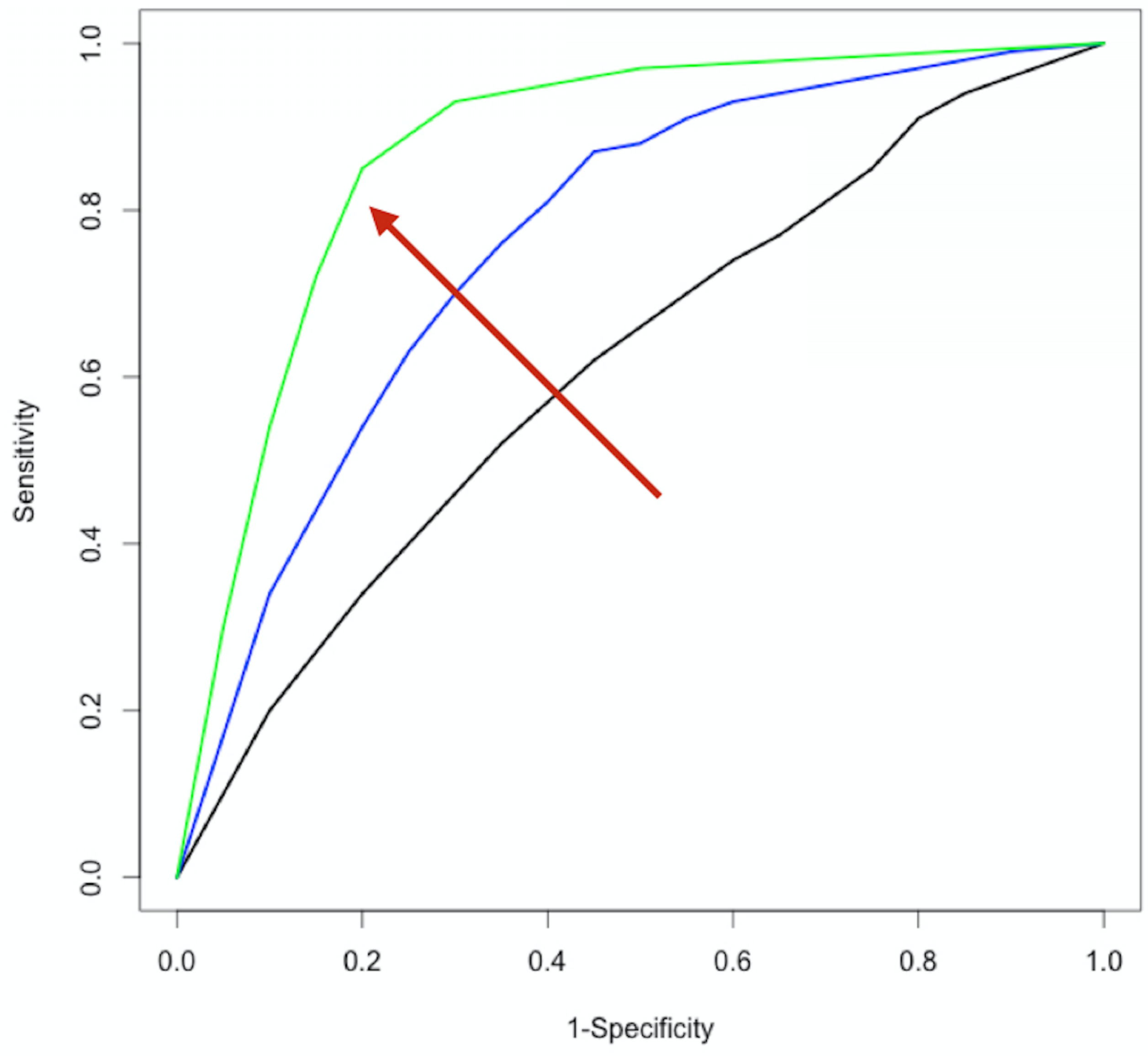
The ROC-curve



$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

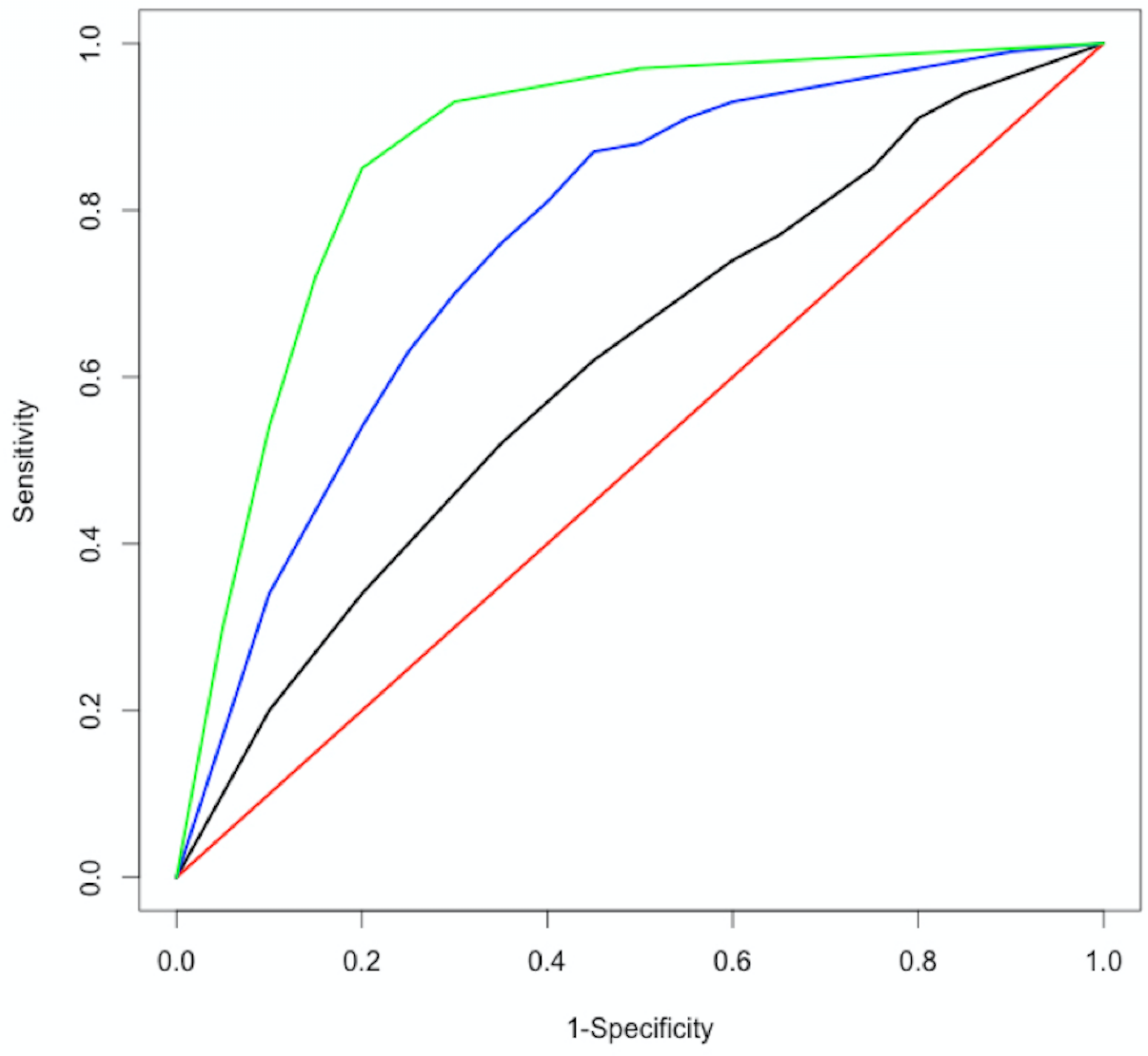
The ROC-curve



$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

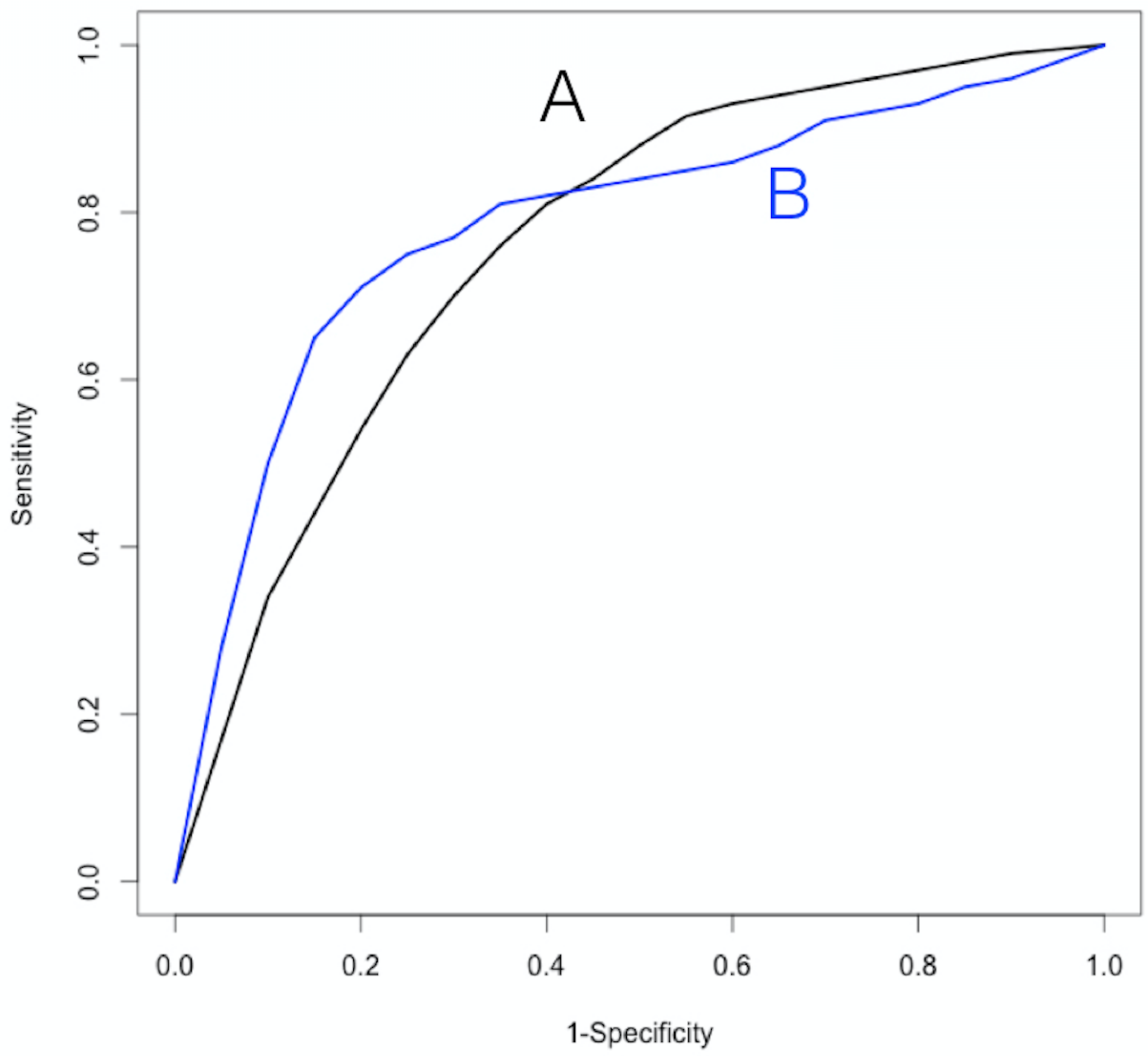
The ROC-curve



$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

Which one is better?



- AUC ROC-curve `A = 0.75`
- AUC ROC-curve `B = 0.78`

Let's practice!

CREDIT RISK MODELING IN R

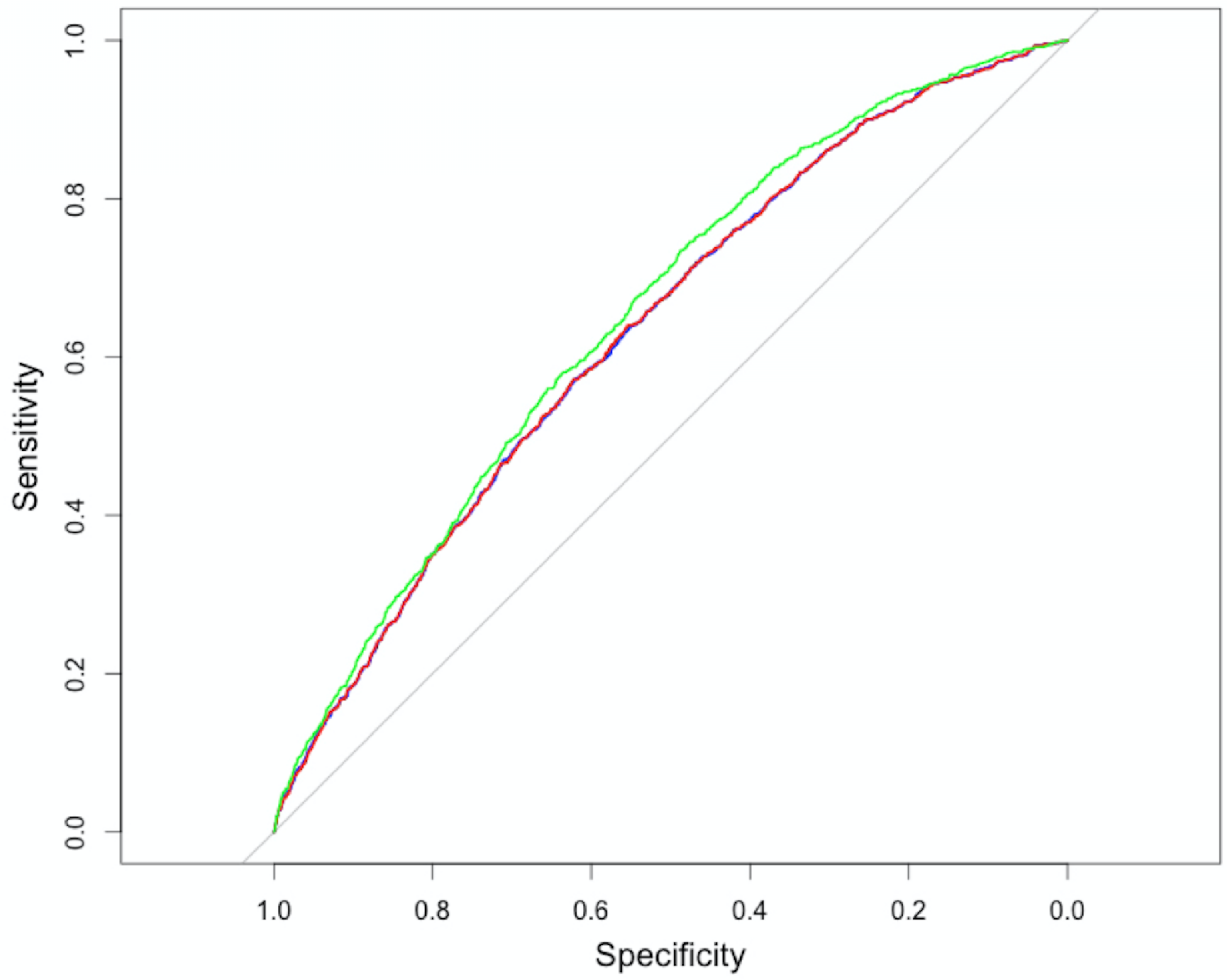
Input selection based on the AUC

CREDIT RISK MODELING IN R

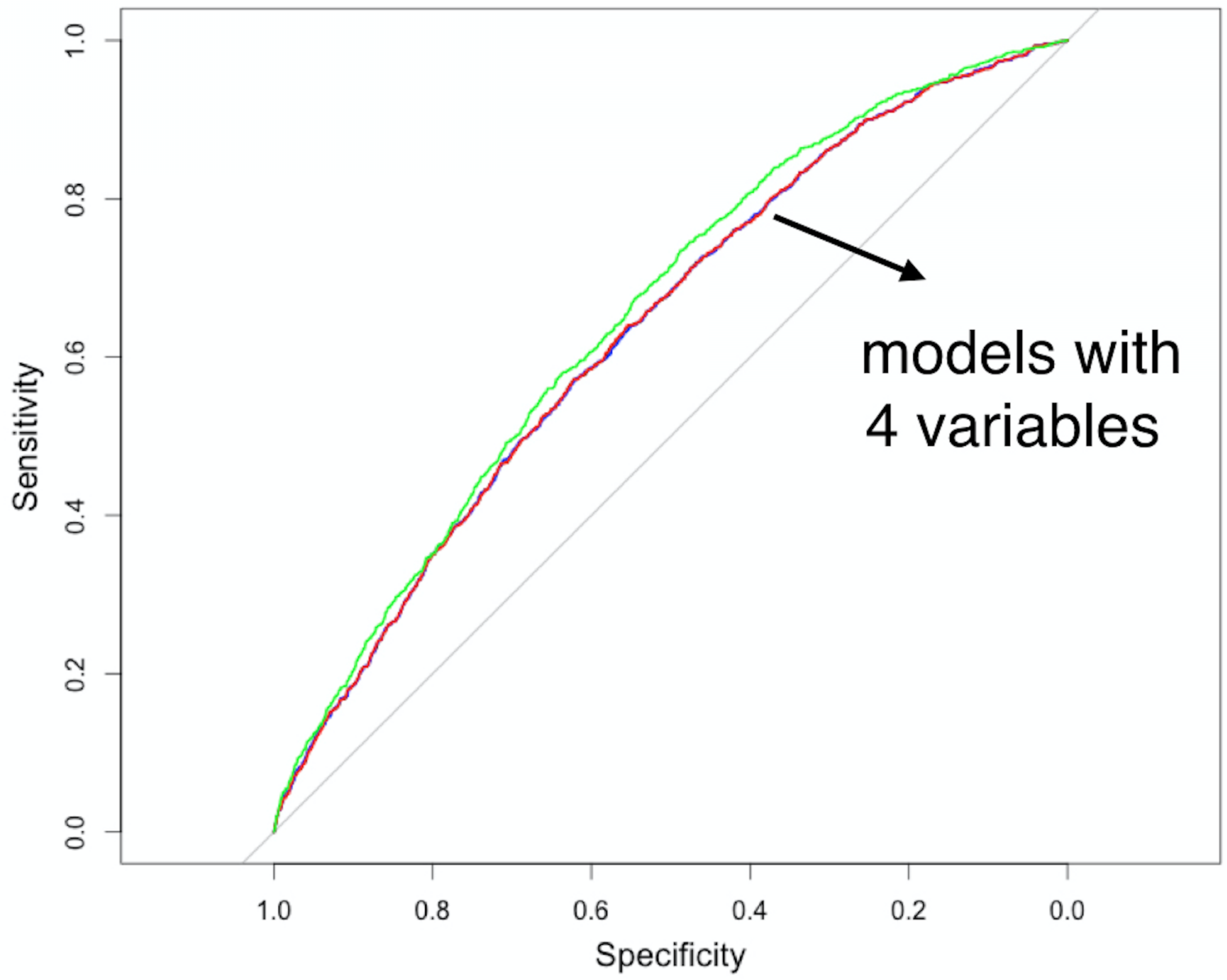
Lore Dirick

Manager of Data Science Curriculum at Flatiron School

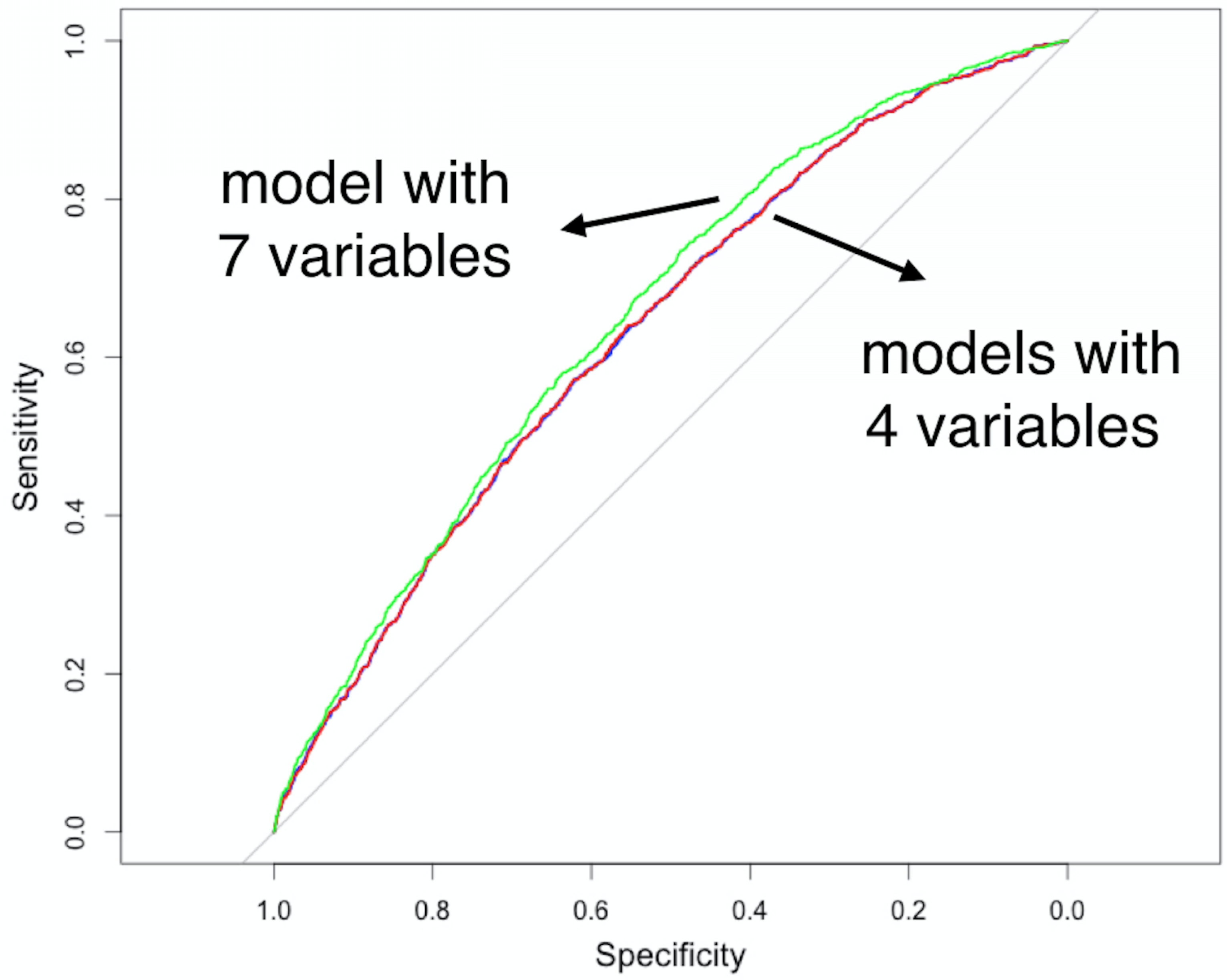
ROC curves for 4 logistic regression models



ROC curves for 4 logistic regression models



ROC curves for 4 logistic regression models



AUC-based pruning

1) Start with a model including all variables (in our case, 7) and compute AUC

```
log_model_full <- glm(loan_status ~ loan_amnt + grade + home_ownership +
                      annual_inc + age + emp_cat + ir_cat,
                      family = "binomial", data = training_set)

predictions_model_full <- predict(log_model_full,
                                  newdata = test_set, type ="response")

AUC_model_full <- auc(test_set$loan_status, predictions_model_full)
```

```
Area under the curve: 0.6512
```

2) Build 7 new models, where each time one of the variables is removed, and make PD-predictions using the test set

```
log_1_remove_amnt <- glm(loan_status ~ grade + home_ownership + annual_inc + age + emp_cat + ir_cat,
                         family = "binomial",
                         data = training_set)

log_1_remove_grade <- glm(loan_status ~ loan_amnt + home_ownership + annual_inc + age + emp_cat + ir_cat,
                          family = "binomial",
                          data = training_set)

log_1_remove_home <- glm(loan_status ~ loan_amnt + grade + annual_inc + age + emp_cat + ir_cat,
                         family = "binomial",
                         data = training_set)

pred_1_remove_amnt <- predict(log_1_remove_amnt, newdata = test_set, type = "response")
pred_1_remove_grade <- predict(log_1_remove_grade, newdata = test_set, type = "response")
pred_1_remove_home <- predict(log_1_remove_home, newdata = test_set, type = "response")
...
```

3) Keep the model that led to the best AUC (AUC full model: 0.6512)

```
auc(test_set$loan_status, pred_1_remove_amnt)
```

```
Area under the curve: 0.6537
```

```
auc(test_set$loan_status, pred_1_remove_grade)
```

```
Area under the curve: 0.6438
```

```
auc(test_set$loan_status, pred_1_remove_home)
```

```
Area under the curve: 0.6537
```

4) Repeat until AUC decreases (significantly)

Let's practice!

CREDIT RISK MODELING IN R

Course wrap-up

CREDIT RISK MODELING IN R

Lore Dirick

Manager of Data Science Curriculum at Flatiron School

Other methods

- Discriminant analysis
- Random forest
- Neural networks
- Support vector machines

But... very classification-focused

- Timing aspect is neglected
- New popular method: survival analysis
  - PDs that change over time
  - Time-varying covariates can be included

Congratulations!

CREDIT RISK MODELING IN R