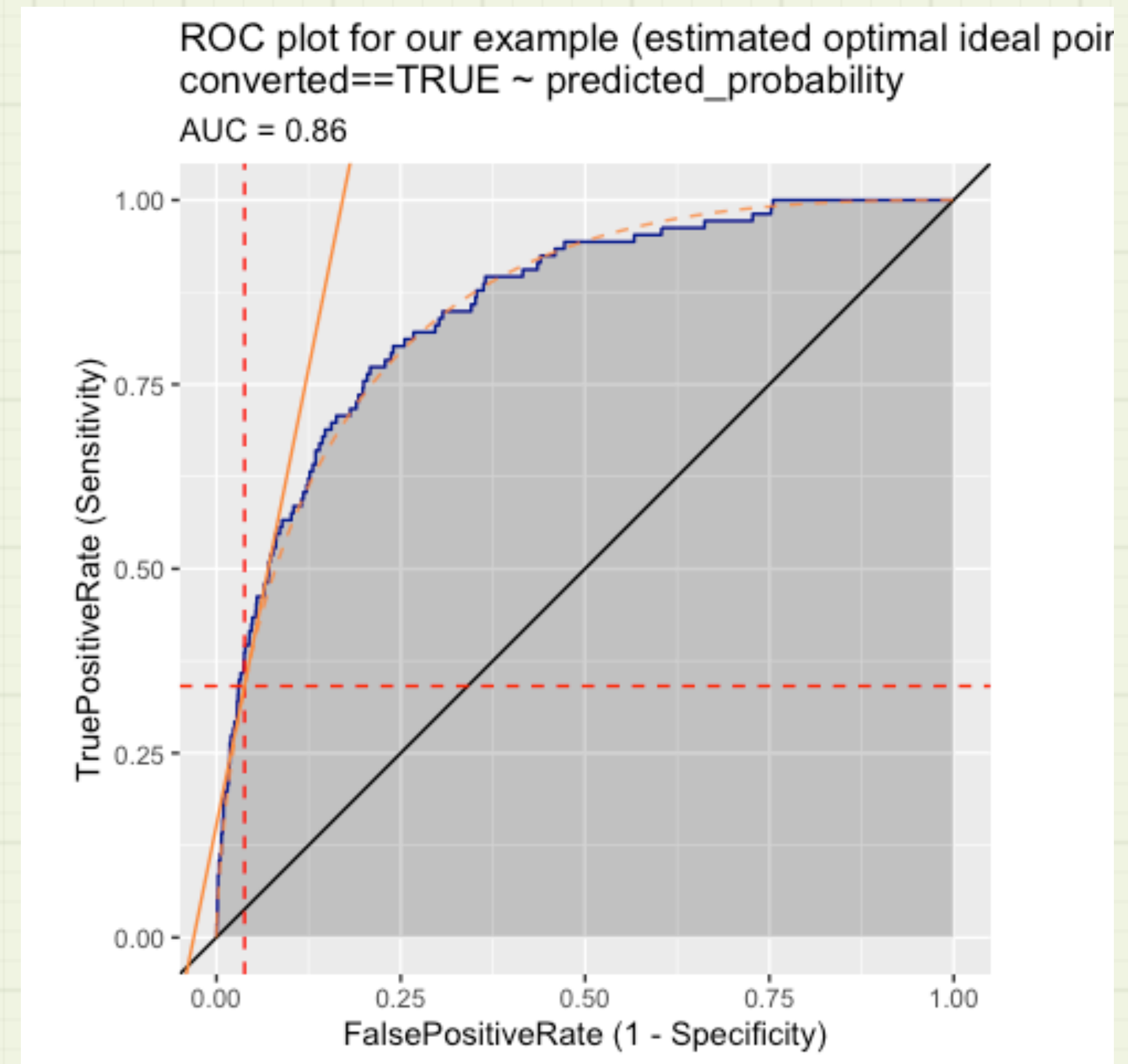


Squeezing the Most Utility from Your Models

Nina Zumel
Win-Vector LLC

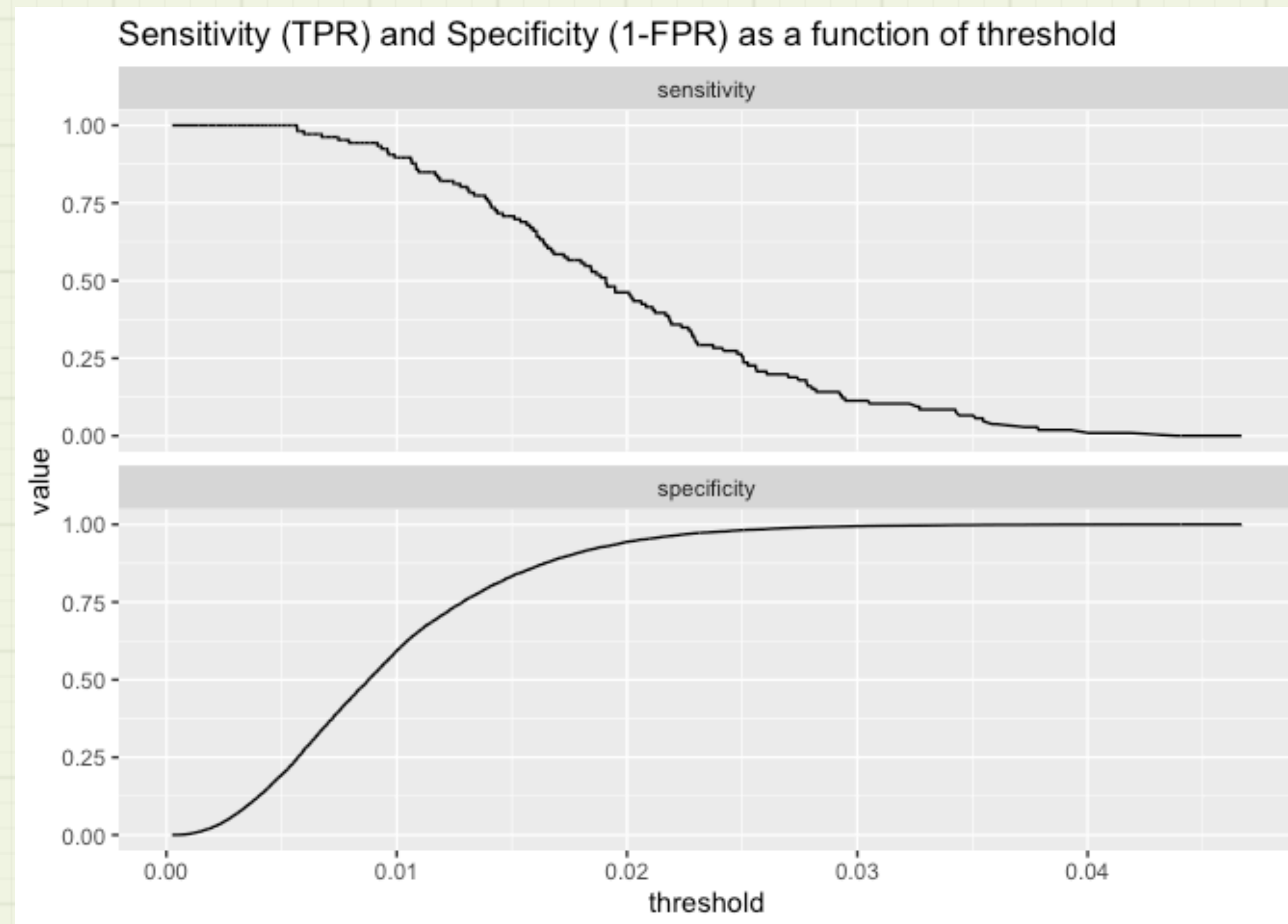
Picking Thresholds: An Alternative View

- ROCs are great for visualizing certain tradeoffs, but...
- Thresholds only implicit in the visualization
- Harder to visualize other trade-offs
 - Precision/Recall
 - Revenue vs Expenditure



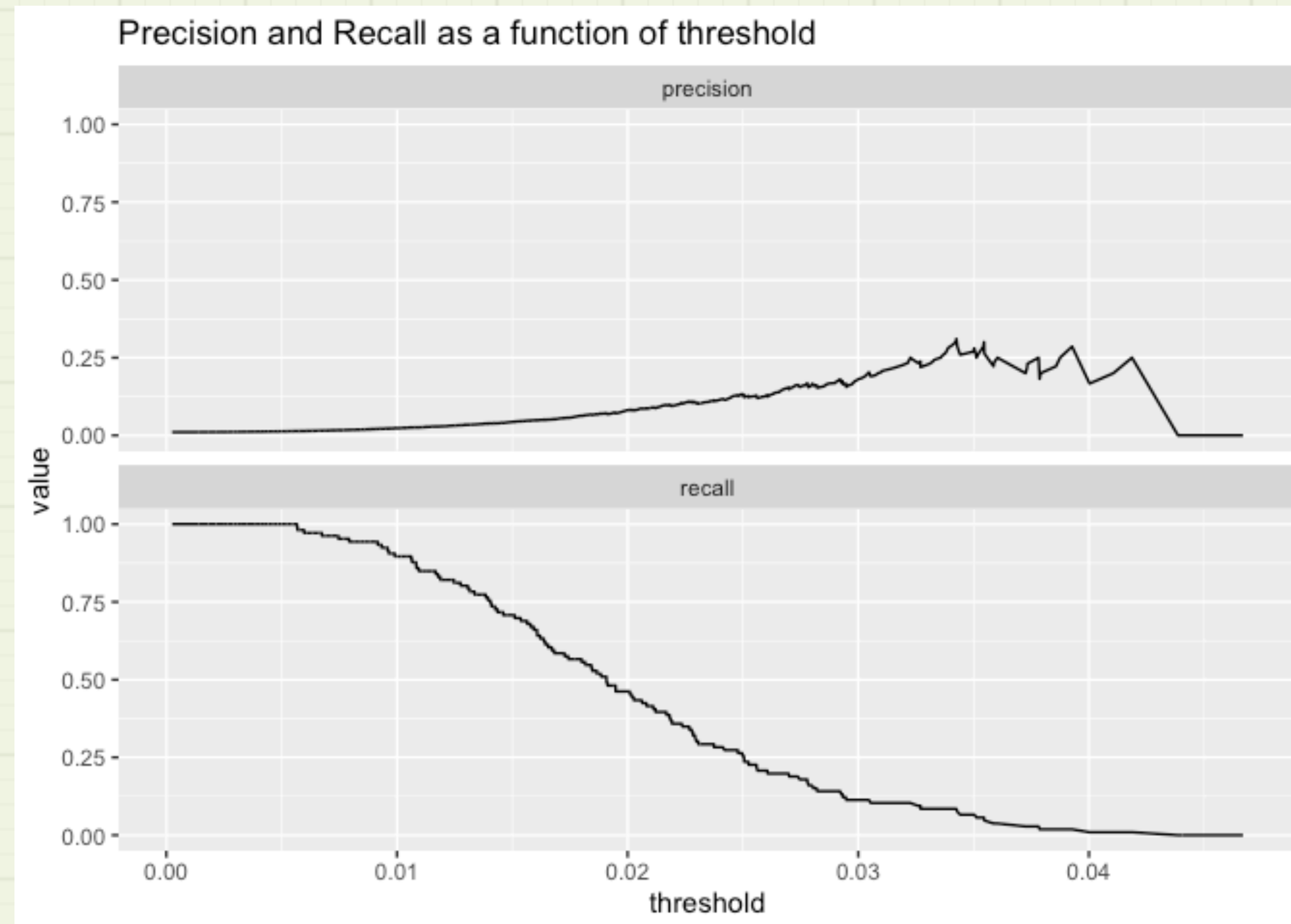
"Unroll" the ROC

```
WVPlots::ThresholdPlot(d, "predicted_probability", "converted",  
  title = "Sensitivity (TPR) and Specificity (1-FPR) as a function of threshold")
```



Tradeoffs on other Metrics

```
ThresholdPlot(d, "predicted_probability", "converted",  
             metrics =c("precision", "recall"),  
             title = "Precision and Recall as a function of threshold")
```



Model Performance Goals are *not* Operational Goals

- Models help make decisions that run the organization
- Accuracy metrics aren't always easy to map to operational goals



Pick Thresholds By Model *Utility*

- Rewards/costs for correct/incorrect classifications
- Assign values to TP, FP, TN, FN
 - (Generally monetary)
- (How thresholds picked via ROC)



Example 1: Sales

Identify and contact prospective customers

- Pool of 10,000 prospects
 - Conversion rate 1% (~100 customers)
- Every contact costs $\$x_{\text{contact}}$
- Every conversion is worth $\$x_{\text{sale}}$

Goal: Maximize booked net revenue



Example 1: Sales

Identify and contact prospective customers

- Pool of 10,000 customers
 - Conversion rate 1% (~100 customers)
- Every contact costs $\$x_{\text{contact}}$
- Every conversion is worth $\$x_{\text{sale}}$

Goal: Maximize booked net revenue

Utilities

true positive value: $x_{\text{sale}} - x_{\text{contact}}$
(prospects contacted and converted)

Example: Sales

Identify and contact prospective customers

- Pool of 10,000 customers
 - Conversion rate 1% (~100 customers)
- Every contact costs $\$x_{\text{contact}}$
- Every conversion is worth $\$x_{\text{sale}}$

Goal: Maximize booked net revenue

Utilities

true positive value: $x_{\text{sale}} - x_{\text{contact}}$

false positive value: $-x_{\text{contact}}$

(prospects contacted who didn't convert)

Example 1: Sales

Identify and contact prospective customers

- Pool of 10,000 customers
 - Conversion rate 1% (~100 customers)
- Every contact costs $\$X_{\text{contact}}$
- Every conversion is worth $\$X_{\text{sale}}$

Goal: Maximize booked net revenue

Utilities

true positive value: $X_{\text{sale}} - X_{\text{contact}}$

false positive value: $-X_{\text{contact}}$

true negative value: 0

(not contacted, wouldn't have converted)

false negative value: -0.01

(not contacted, would have converted)

Example 2: Medical Screening

Screening for a rare but serious condition

- ~1% prevalence (~100 out of 10,000 subjects)
- Test costs $\$T$
- Follow-up costs $\$F$
- Early treatment costs $\$x_{\text{early_treat}}$
- Late treatment costs $\$x_{\text{late_treat}} \gg \$x_{\text{early_treat}}$

Goal: minimize overall cost
= maximize (negative) value



Example 2: Medical Screening

Screening for a rare but serious condition

- ~1 % prevalence (~100 out of 10,000 subjects)
- Test costs $\$T$
- Follow-up costs $\$F$
- Early treatment costs $\$X_{\text{early_treat}}$
- Late treatment costs $\$X_{\text{late_treat}} \gg \$X_{\text{early_treat}}$

**Goal: minimize overall cost
= maximize (negative) value**

Utilities

true positive value: $-(T + F + X_{\text{early_treat}})$
(subject caught early)

Example 2: Medical Screening

Screening for a rare but serious condition

- ~1 % prevalence (~100 out of 10,000 subjects)
- Test costs $\$T$
- Follow-up costs $\$F$
- Early treatment costs $\$X_{\text{early_treat}}$
- Late treatment costs $\$X_{\text{late_treat}} \gg \$X_{\text{early_treat}}$

**Goal: minimize overall cost
= maximize (negative) value**

Utilities

true positive value: $-(T + F + X_{\text{early_treat}})$

false positive value: $-(T + F)$

(subjects who mistakenly screen positive)

Example 2: Medical Screening

Screening for a rare but serious condition

- ~1 % prevalence (~100 out of 10,000 subjects)
- Test costs $\$T$
- Follow-up costs $\$F$
- Early treatment costs $\$X_{\text{early_treat}}$
- Late treatment costs $\$X_{\text{late_treat}} \gg \$X_{\text{early_treat}}$

**Goal: minimize overall cost
= maximize (negative) value**

Utilities

true positive value: $-(T + F + X_{\text{early_treat}})$

false positive value: $-(T + F)$

true negative value: $-T$

(subjects who correctly screen negative)

Example 2: Medical Screening

Screening for a rare but serious condition

- ~1 % prevalence (~100 out of 10,000 subjects)
- Test costs $\$T$
- Follow-up costs $\$F$
- Early treatment costs $\$X_{\text{early_treat}}$
- Late treatment costs $\$X_{\text{late_treat}} \gg \$X_{\text{early_treat}}$

**Goal: minimize overall cost
= maximize (negative) value**

Utilities

true positive value: $-(T + F + X_{\text{early_treat}})$

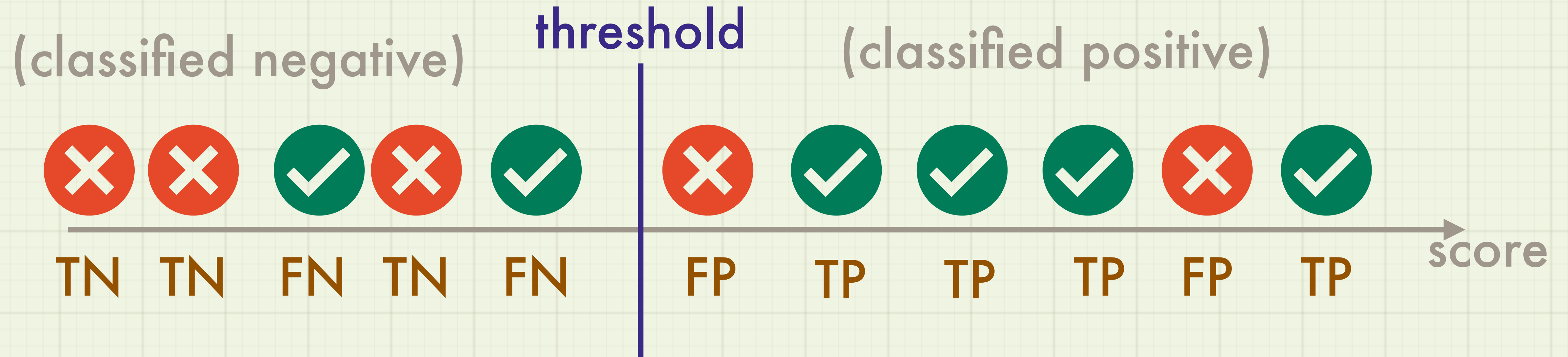
false positive value: $-(T + F)$

true negative value: $-T$

false negative value: $-(T + X_{\text{late_treat}})$

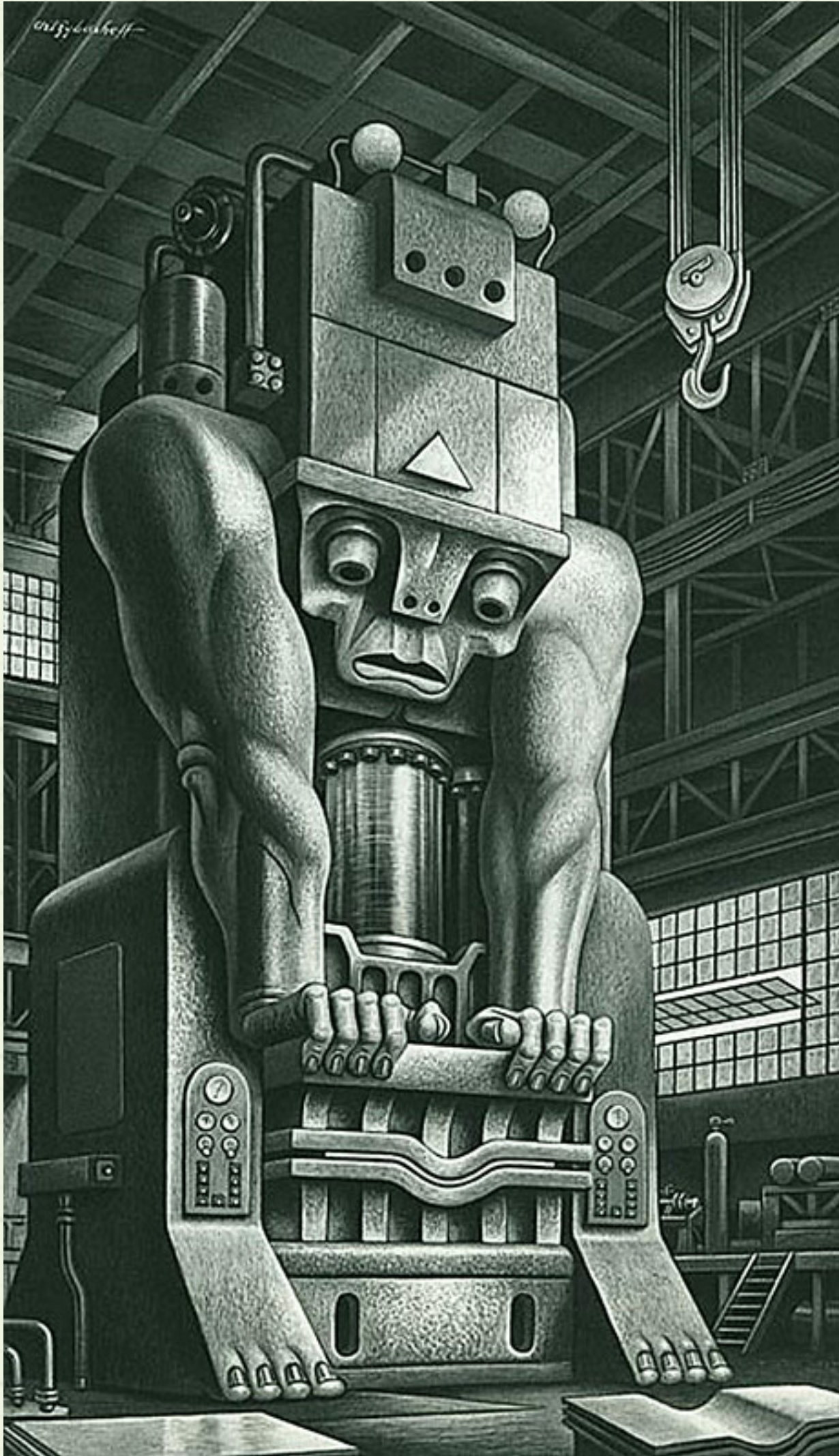
(subjects we missed in early screening)

Calculating Total Utility



$$\text{utility} = 3 * \text{TN}_v + 2 * \text{FN}_v + 2 * \text{FP}_v + 4 * \text{TP}_v$$

Optimal threshold: maximizes total utility



Let's Do It!

Back to Sales Example

`head(d)`

converted	predicted_probability
FALSE	0.0040164
FALSE	0.0199652
FALSE	0.0132867
FALSE	0.0051605
FALSE	0.0038753
FALSE	0.0057591

cost of call: \$5
value of sale: \$100
missed opportunity: -\$0.01

```
(prevalence = mean(d$converted))  
## [1] 0.0106
```

```
d$true_positive_value <- 100 - 5  
                        # conversion - cost  
d$false_positive_value <- -5  
                        # the cost of a call  
d$true_negative_value <- 0  
d$false_negative_value <- -0.01  
                        # a small penalty
```


Calculate model utility

```
library(sigr)
values <- model_utility(
  d,
  model_name = 'predicted_probability',
  # column of scores
  outcome_name = 'converted')
# actual outcomes
```

`sigr::model_utility()` returns
a data frame of utility calculations
as a function of threshold

Calculate model utility

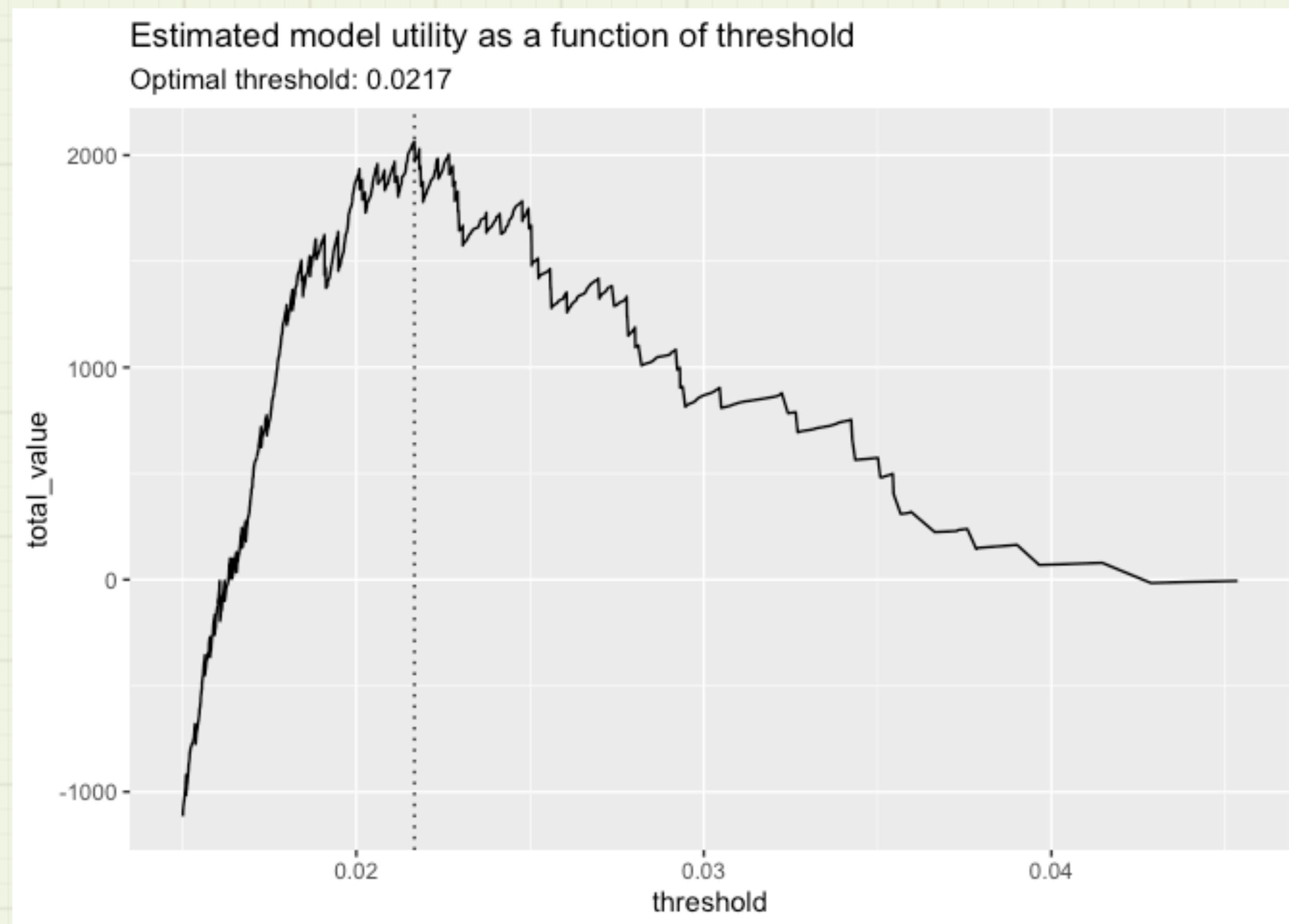
```
library(sigr)
values <- model_utility(
  d,
  model_name = 'predicted_probability',
  # column of scores
  outcome_name = 'converted')
# actual outcomes
```

threshold: 0.015

- contact 1751 prospects (of 10,000)
- 75 conversions (of 106 possible)
- earned \$7125, spent \$8380
- **total utility -\$1255.31**

model	predicted_proba bility
threshold	0.01493523
count_taken	1751
fraction_taken	0.1751
true_positive_value	7125
false_positive_value	-8380
true_negative_value	0
false_negative_value	-0.31
total_value	-1255.31
true_negative_count	8218
false_negative_count	31
true_positive_count	75
false_positive_count	1676

Utility as a function of threshold



threshold	0.021672
count_taken	427.000000
true_positive_count	42.000000
false_positive_count	385.000000
true_negative_count	9509.000000
false_negative_count	64.000000
fraction_taken	0.042700
total_value	2064.360000

Comparing to Best Possible Performance

Our model realized $2064/10070 = 20.1\%$ of best possible

```
# add a column for the wizard model
d$wizard <- with(d, ifelse(converted, 1.0, 0.0))

# calculate the wizard's model utilities
wizard_values <- model_utility(d,
                               model_name = 'wizard',
                               outcome_name = 'converted')
```

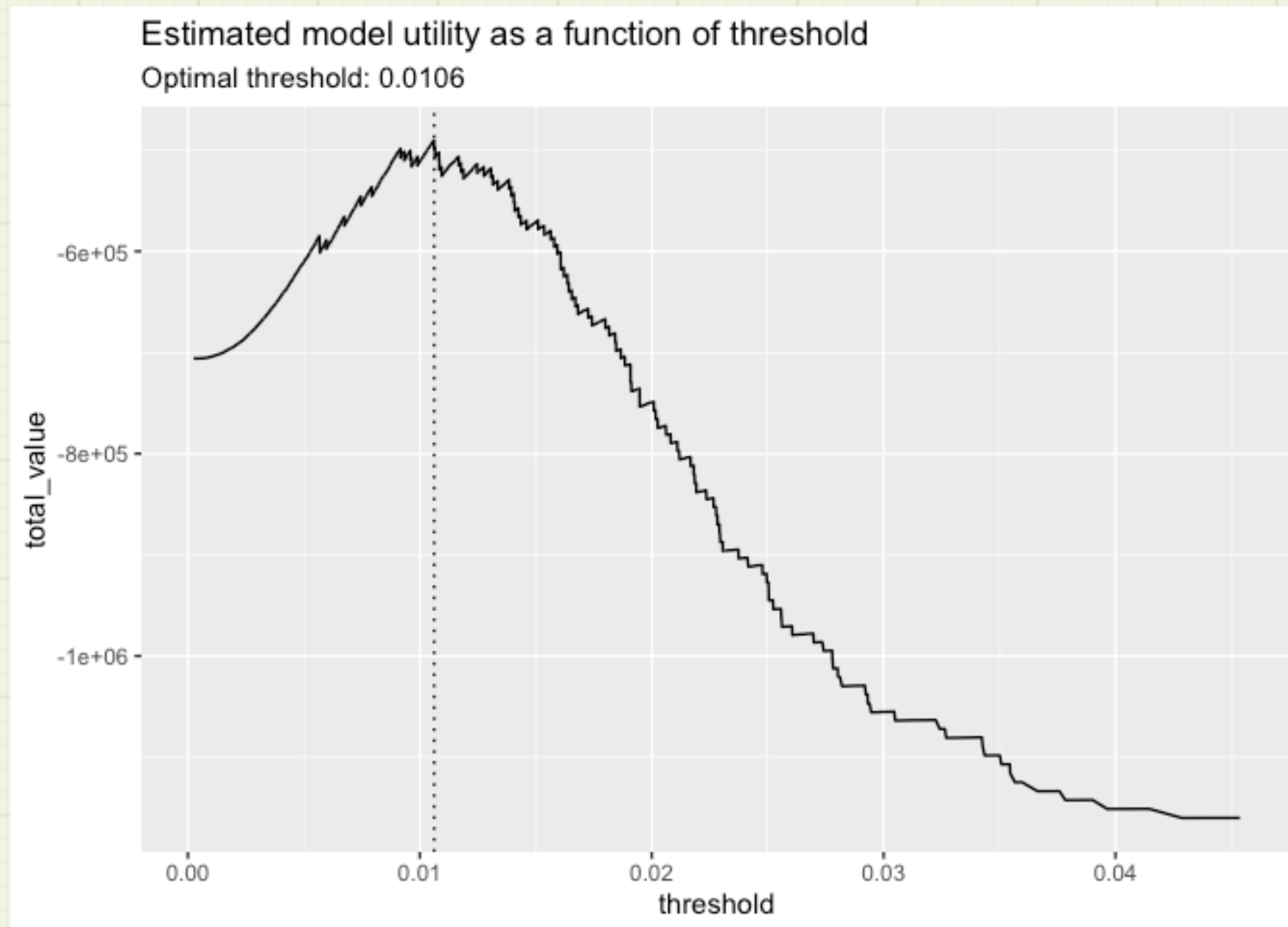
threshold	count_taken	fraction_taken	total_value	
0.0	10000	1.0000	-39400.00	(contact everyone)
0.5	106	0.0106	10070.00	(contact exactly the right prospects)
NA	0	0.0000	-1.06	(contact no one)

Medical Screening Example

```
x_test = 10  
x_follow_up = 50  
x_early_treatment = 1000  
x_late_treatment = 10000
```

```
d$true_positive_value <- -(x_test + x_follow_up + x_early_treatment)  
d$false_positive_value <- -(x_test + x_follow_up)  
d$true_negative_value <- -x_test  
d$false_negative_value <- -(x_test + x_late_treatment)
```

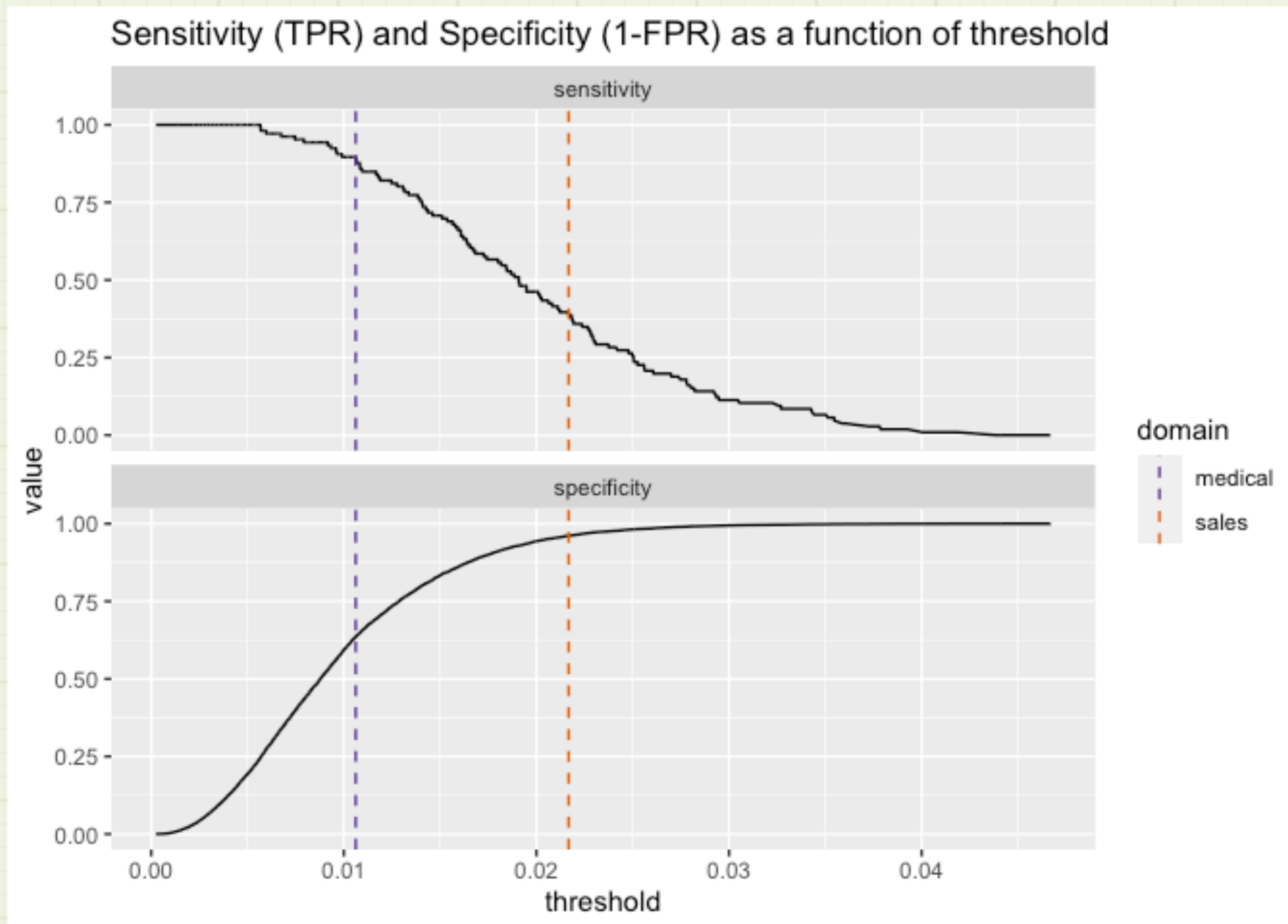
Find the Best Threshold



threshold	1.06178E-02
count_taken	3704
true_positive_count	95
false_positive_count	3609
true_negative_count	6285
false_negative_count	11
fraction_taken	0.03704
total_value	-490200

Lowest possible cost: \$211,300

Same Model Performance, Different Operational Goals



Medical:
High sensitivity important
False Negatives expensive

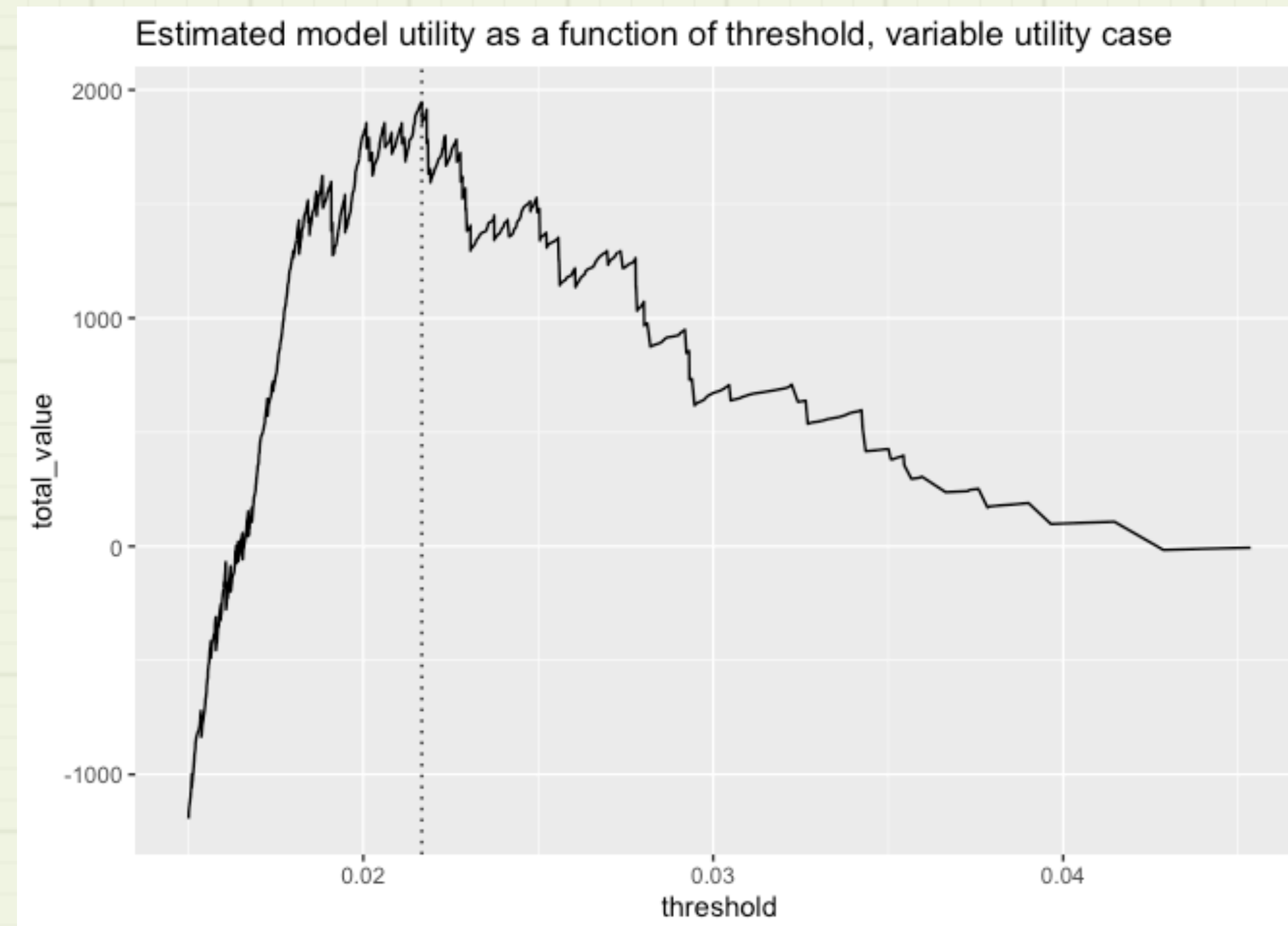
Sales:
Higher specificity preferred
False Positives add up

Advanced: Variable Utilities

```
d$false_positive_value <- -5          # the cost of a call
d$true_negative_value <- 0
d$false_negative_value <- -0.01      # a small penalty

# replace the true positive value with a varying value
d$true_positive_value <- rnorm(nrow(d), mean = 100, sd = 25) - 5 # revenue - cost

values <- model_utility(d,
                        model_name = 'predicted_probability',
                        outcome_name = 'converted')
```



Takeaways

- Accuracy metrics don't always express how (or whether) a model meets operational goals
- Utilities express model performance in "operational units" (e.g. \$)
 - Quantify tradeoffs between false positives and missed opportunities
- Somewhat more intuitive way to elicit performance goals from stakeholders and business partners



Thank You