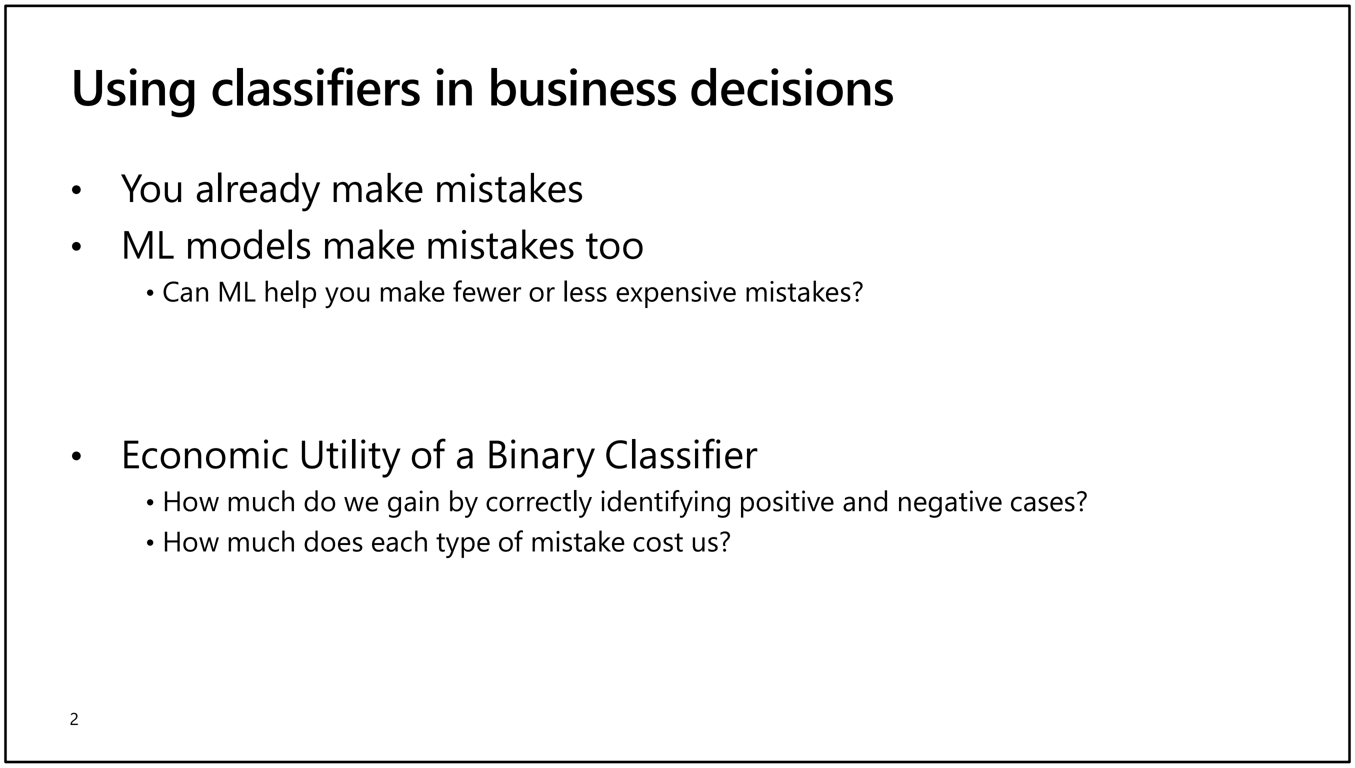
Automating Decisions: Quantifying quality/quantity tradeoffs in ML

Robert Horton

1

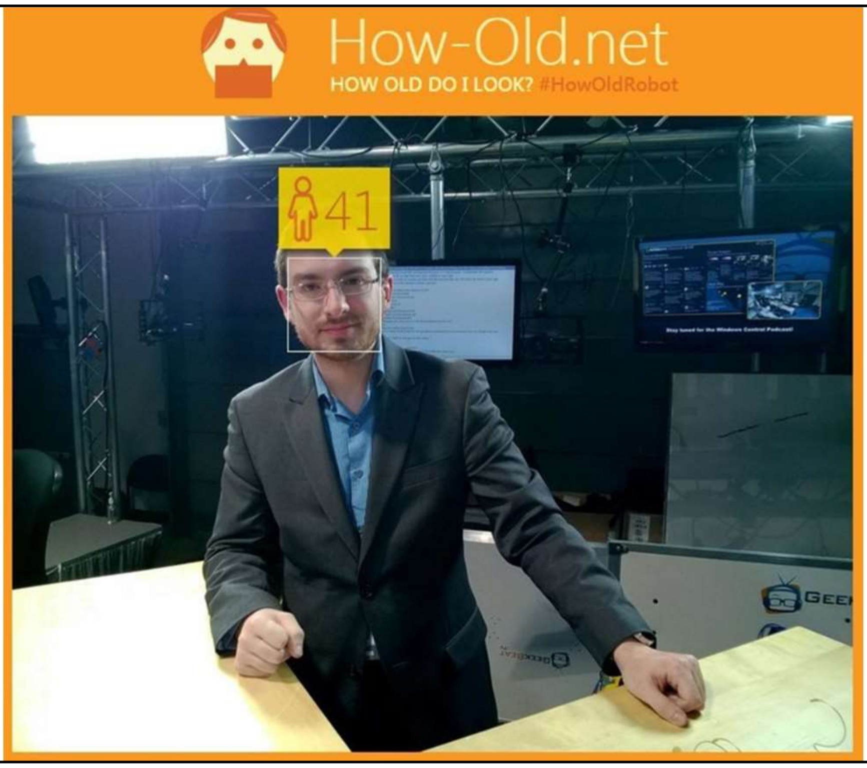


Using classifiers in business decisions

- You already make mistakes
- ML models make mistakes too
  - Can ML help you make fewer or less expensive mistakes?

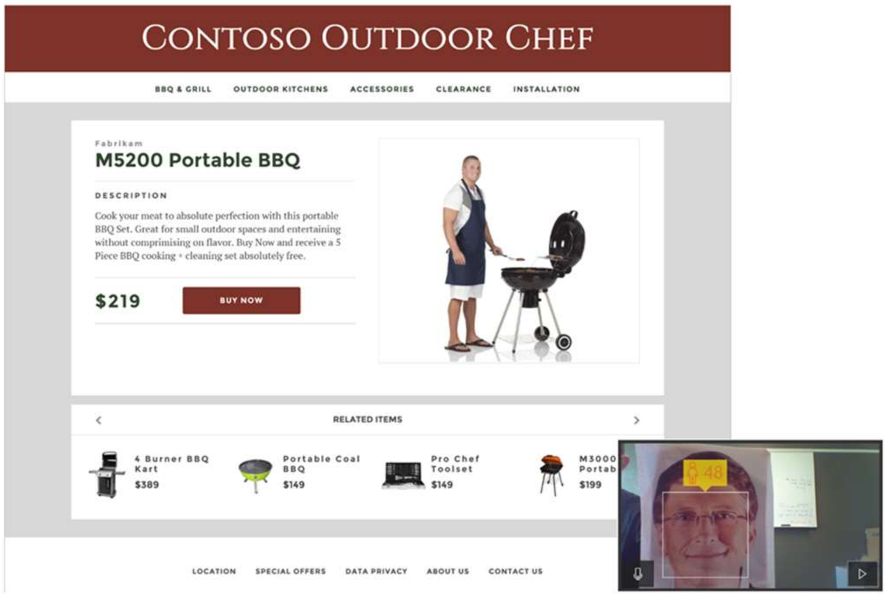
- Economic Utility of a Binary Classifier
  - How much do we gain by correctly identifying positive and negative cases?
  - How much does each type of mistake cost us?

2



According to the windowscentral article, this guy is only 28 years old.
People are not favorably impressed when you get their age or sex wrong.

Mall Kiosk



CONTOSO OUTDOOR CHEF

BBQ & GRILL OUTDOOR KITCHENS ACCESSORIES CLEARANCE INSTALLATION

Fabrikam

M5200 Portable BBQ

DESCRIPTION

Cook your meat to absolute perfection with this portable BBQ Set. Great for small outdoor spaces and entertaining without comprimising on flavor. Buy Now and receive a 5 Piece BBQ cooking + cleaning set absolutely free.

$219 BUY NOW

RELATED ITEMS

4 Burner BBQ Kart $389

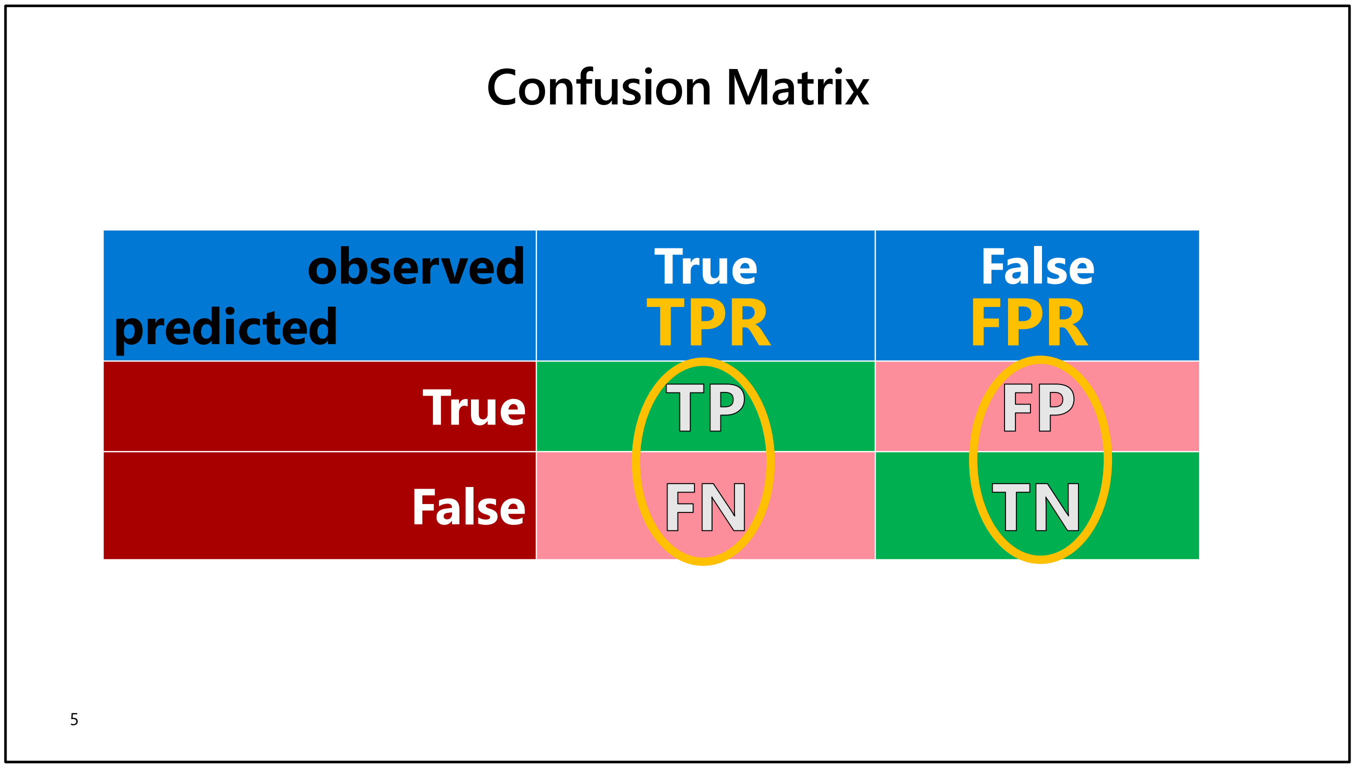
Portable Coal BBQ $149

Pro Chef Toolset $149

M3000 Portab $199

LOCATION SPECIAL OFFERS DATA PRIVACY ABOUT US CONTACT US

The kiosk application uses the same model to estimate age and sex, but it doesn’t just return the estimates; it uses them to choose an ad to show. The cost of estimating incorrectly is much lower in this system.



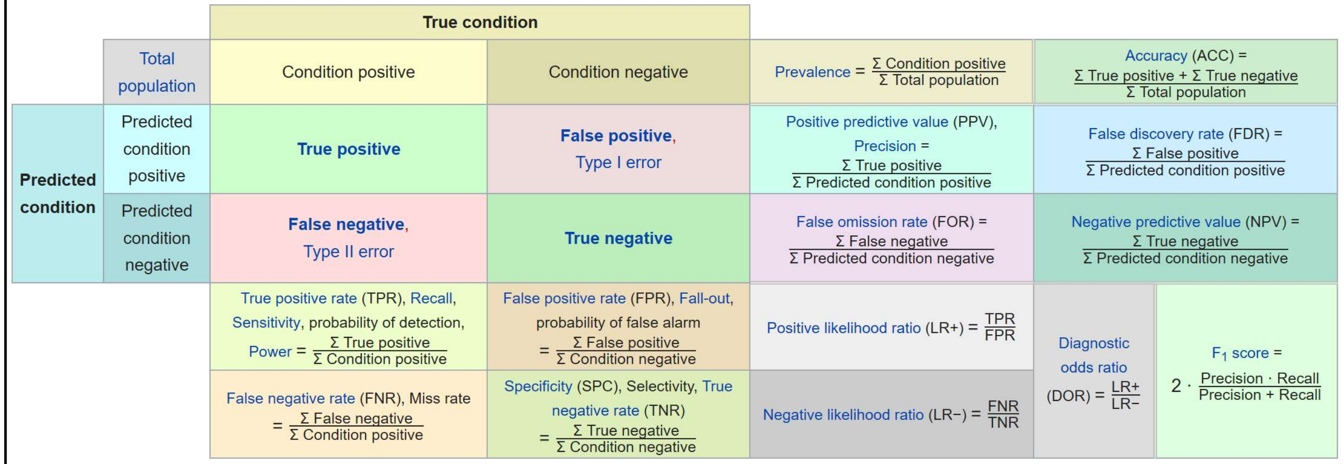
Confusion Matrix

observed predicted	True TPR	False FPR
True	TP	FP
False	FN	TN

5

Confusion matrixes seem like a simple idea. It is a table with only 4 cells, how complicated can it be?

We often use ratios of these cell values to compute useful metrics (eg, precision or sensitivity), which adds a bit of combinatorial complexity. And since these ideas are so important, they have generally been discovered by people working in different fields, so there are multiple common names for many of these ideas (for example, recall is exactly the same thing as sensitivity; one term comes from engineering and the other from medicine.)



		True condition				
	Total population	Condition positive	Condition negative	Prevalence = $\frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	
Predicted condition	Predicted condition positive	**True positive**	**False positive**, Type I error	Positive predictive value (PPV), Precision = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Predicted condition positive}}$	
	Predicted condition negative	**False negative**, Type II error	**True negative**	False omission rate (FOR) = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Predicted condition negative}}$	
		True positive rate (TPR), Recall, Sensitivity, probability of detection, Power = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	Positive likelihood ratio (LR+) = $\frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR−}}$	F_1 score = $2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$
		False negative rate (FNR), Miss rate = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$	Negative likelihood ratio (LR−) = $\frac{\text{FNR}}{\text{TNR}}$		

https://en.wikipedia.org/wiki/Confusion_matrix

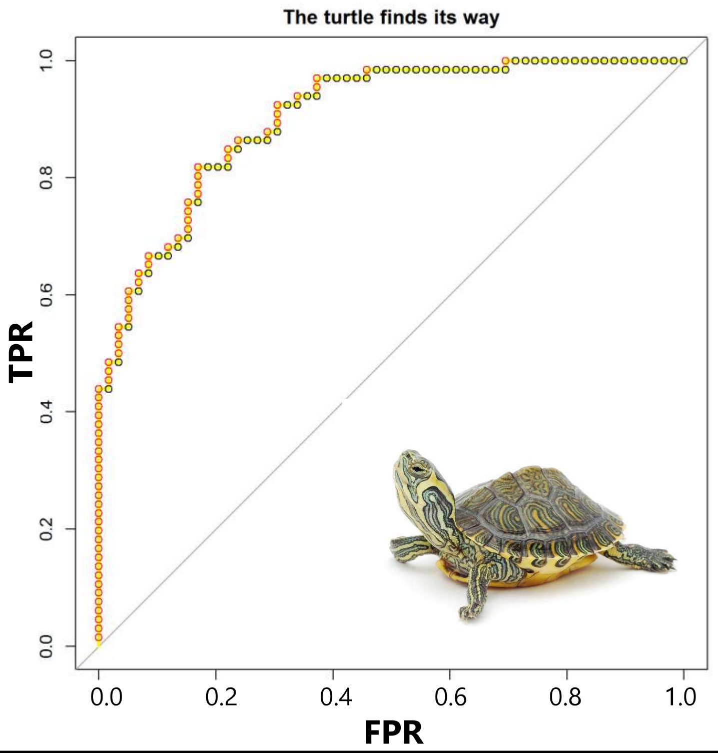
This is a richer presentation of the complexities of a 2 by 2 confusion matrix:
https://en.wikipedia.org/wiki/Confusion_matrix

ROC Curve:

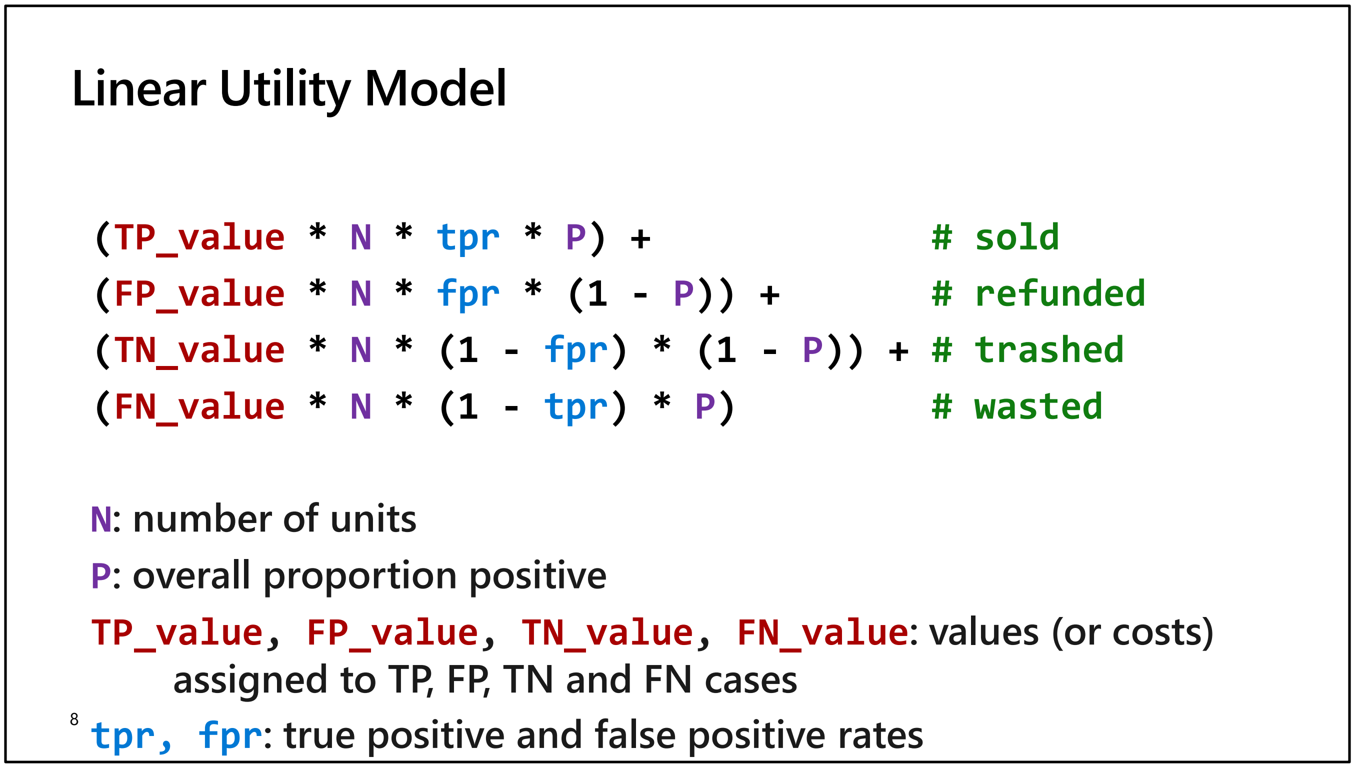
- sort test cases by their score from the model
- march along the sequence, stepping up for positives and right for negatives

This is the same as scanning across possible cutoff threshold values.

The slope of the curve shows the concentration of positives.



The “turtle’s eye view” of ROC curves is described here:
https://blog.revolutionanalytics.com/2016/08/roc-curves-in-two-lines-of-code.html



Linear Utility Model

```
(TP_value * N * tpr * P) +            # sold
(FP_value * N * fpr * (1 - P)) +      # refunded
(TN_value * N * (1 - fpr) * (1 - P)) + # trashed
(FN_value * N * (1 - tpr) * P)        # wasted
```

`N`: number of units

`P`: overall proportion positive

`TP_value`, `FP_value`, `TN_value`, `FN_value`: values (or costs) assigned to TP, FP, TN and FN cases

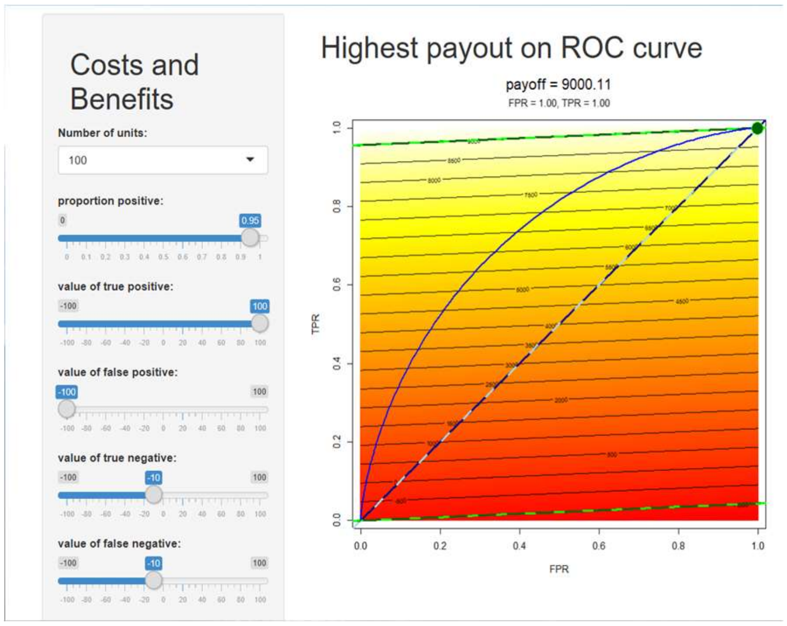
8 `tpr`, `fpr`: true positive and false positive rates

You can make a simple cost/benefit model by assigning a fixed positive or negative value to each of the 4 cells in the confusion matrix.

For a detailed exposition of this approach, complete with use case and a cast of characters, see this excellent blog post by Nicolas Kruchten:
http://blog.mldb.ai/blog/posts/2016/01/ml-meets-economics/

Economic Utility in ROC Space

The settings on the left are used to compute the “payoff” value of every point on the plane of true positive rate (TPR) vs. false positive rate (FPR); this is used to color the background. The parallel black lines are “lines of indifference”, showing contours on the cost surface (since this cost model is linear, these are parallel lines). The ROC curve is shown in blue, and the green spot shows the point on the curve with the highest payoff. In this case the highest payoff is at [1.0, 1.0], where the threshold is set so low that all cases are considered positive, and the classifier is not useful.



Costs and Benefits

Number of units: 100

proportion positive: 0.95

value of true positive: 100

value of false positive: -100

value of true negative: -10

value of false negative: -10

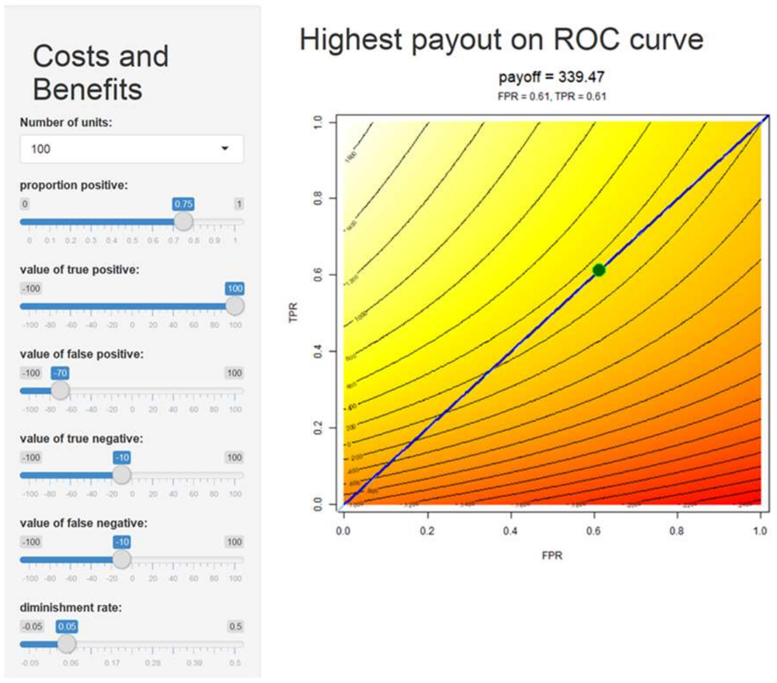
Highest payout on ROC curve

payoff = 9000.11

FPR = 1.00, TPR = 1.00

https://ml4managers.shinyapps.io/ML_utility/

Non-linear utility function



Costs and Benefits

Number of units: 100

proportion positive: 0.75

value of true positive: 100

value of false positive: -70

value of true negative: -10

value of false negative: -10

diminishment rate: 0.05

Highest payout on ROC curve

payoff = 339.47

FPR = 0.61, TPR = 0.61

You can get much more complex payoff surfaces with nonlinear models. Here we get curves by modeling diminishing returns. Other models might have nonlinear step functions, etc.
(for example, your shipping costs might change when you exceed a certain volume).

Resources

- Github repo: https://github.com/microsoft/datascience4managers

- Shiny App:
  - https://ml4managers.shinyapps.io/ML_utility/

- Economic Utility Functions Meet ROC Curves: Deciding on a Cutoff Threshold for Binary Classification. Siddarth Ramesh and Robert Horton, MLADS November 14, 2018. https://resnet.microsoft.com/video/4248
  - https://github.com/Azure/utility_functions_in_ROC_space
  - https://ml4managers.shinyapps.io/ML_utility/

11