MODEL ASSESSMENT

Dr. Aric LaBarr
Institute for Advanced Analytics

COMPARING MODELS

Purpose of Modeling

- Statistical models are created for two different purposes – estimation and prediction.
  - **Estimation:** Quantifying the expected change in response associated with predictors (relationships).
  - **Prediction:** Use the model to predict new response.

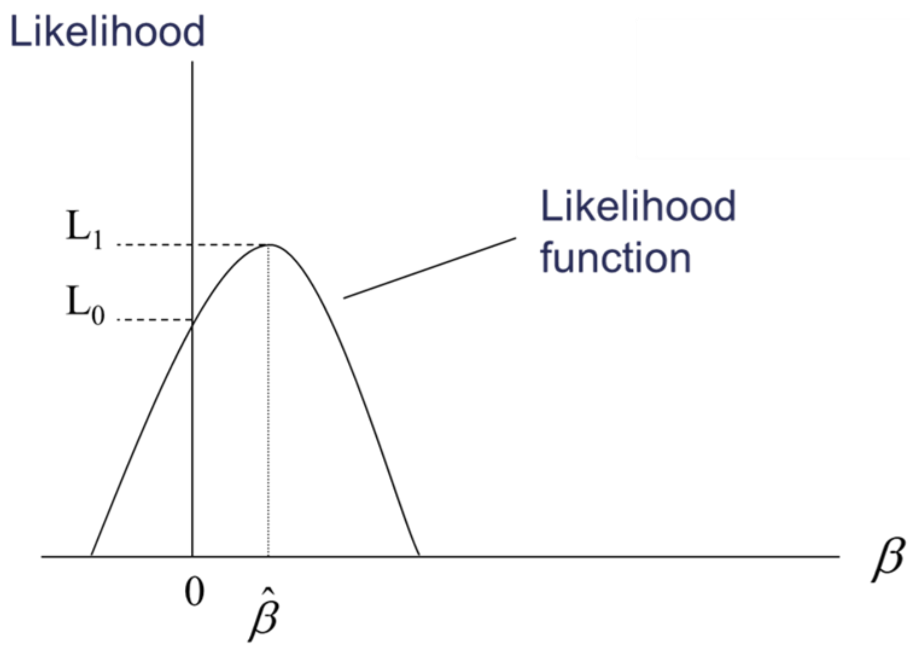
- Won’t necessarily agree!

Deviance/Likelihood Measures

- AIC and BIC approximate out-of-sample prediction error by applying a penalty for model complexity:
  - AIC – crude, large-sample approximation of leave-one-out cross-validation.
  - BIC – favors smaller models/penalizes model complexity more.
- Lower values “better” than higher.
- No amount of lower is “better” enough.
- May not always agree, but neither is necessarily better.

Deviance/Likelihood Measures

- Number of "pseudo"-R^2 quantities for logistic regression.
- Higher values indicate "better" model.
- Generalized / Nagelkerke R^2 - how much better than intercept only model?
- Unlike linear regression, there is **no interpretation** on these.



$$R_G^2 = 1 - \left(\frac{L_0}{L_1}\right)^{\frac{2}{n}}$$

Generalized R^2 – SAS

```
proc logistic data=logistic.lowbwt plots(only)=(oddsratio);
   class race(ref='white') / param=ref;
   model low(event='1') = race lwt smoke /
                          rsq clodds=pl clparm=pl;
   title 'Modeling Low Birth Weight';
run;
quit;
```

Generalized R^2 – SAS

Model Fit Statistics		
Criterion	**Intercept Only**	**Intercept and Covariates**
AIC	236.672	225.015
SC	239.914	241.223
-2 Log L	234.672	215.015

R-Square	0.0988	**Max-rescaled R-Square**	0.1389

Generalized R^2 – R

```
logit.model <- glm(low ~ lwt + factor(smoke) + factor(race),
                   data = bwt, family = binomial(link = "logit"))
summary(logit.model)
```

Generalized R^2 – R

```
AIC(logit.model)

## [1] 225.0147

BIC(logit.model)

## [1] 241.2234

PseudoR2(logit.model, which = "Nagelkerke")

## Nagelkerke
##  0.1389144
```

ASSESSING PREDICTIVE POWER

What is a Good Logistic Model?

- Logistic regression is a **model for probability of an event** – NOT the occurrence of an event.
- Logistic regression **can** be a classification model as well.

- Good model should reflect both of these, but importance of one over the other depends on the problem.

Discrimination vs. Calibration

- **Discrimination** – ability to separate the events from the non-events. How good is model at distinguishing the 1's from the 0's.
- **Calibration** – how well predicted probabilities agree with the actual frequency of the outcomes. Are predicted probabilities systematically too low/high?

- **May not agree with each other!**

ASSESSING PREDICTIVE POWER

Probability Based Metrics

Coefficient of Discrimination

- Want model to assign a higher probability to events and lower probability to non-events.
- **Coefficient of discrimination** (or **discrimination slope**) is the difference in average predicted probability between 1's and 0's:

$$D = \bar{\hat{p}}_1 - \bar{\hat{p}}_0$$

- Able to compare with histograms as well.

Discrimination Slope – SAS

```
proc logistic data=logistic.lowbwt noprint;
   class race(ref='white') / param=ref;
   model low(event='1') = race lwt smoke;
   output out=predprobs p=phat;
run;

proc sort data=predprobs;
   by descending low;
run;

proc ttest data=predprobs order=data;
   ods select statistics summarypanel;
   class low;
   var phat;
   title 'Coefficient of Discrimination and Plots';
run;
```

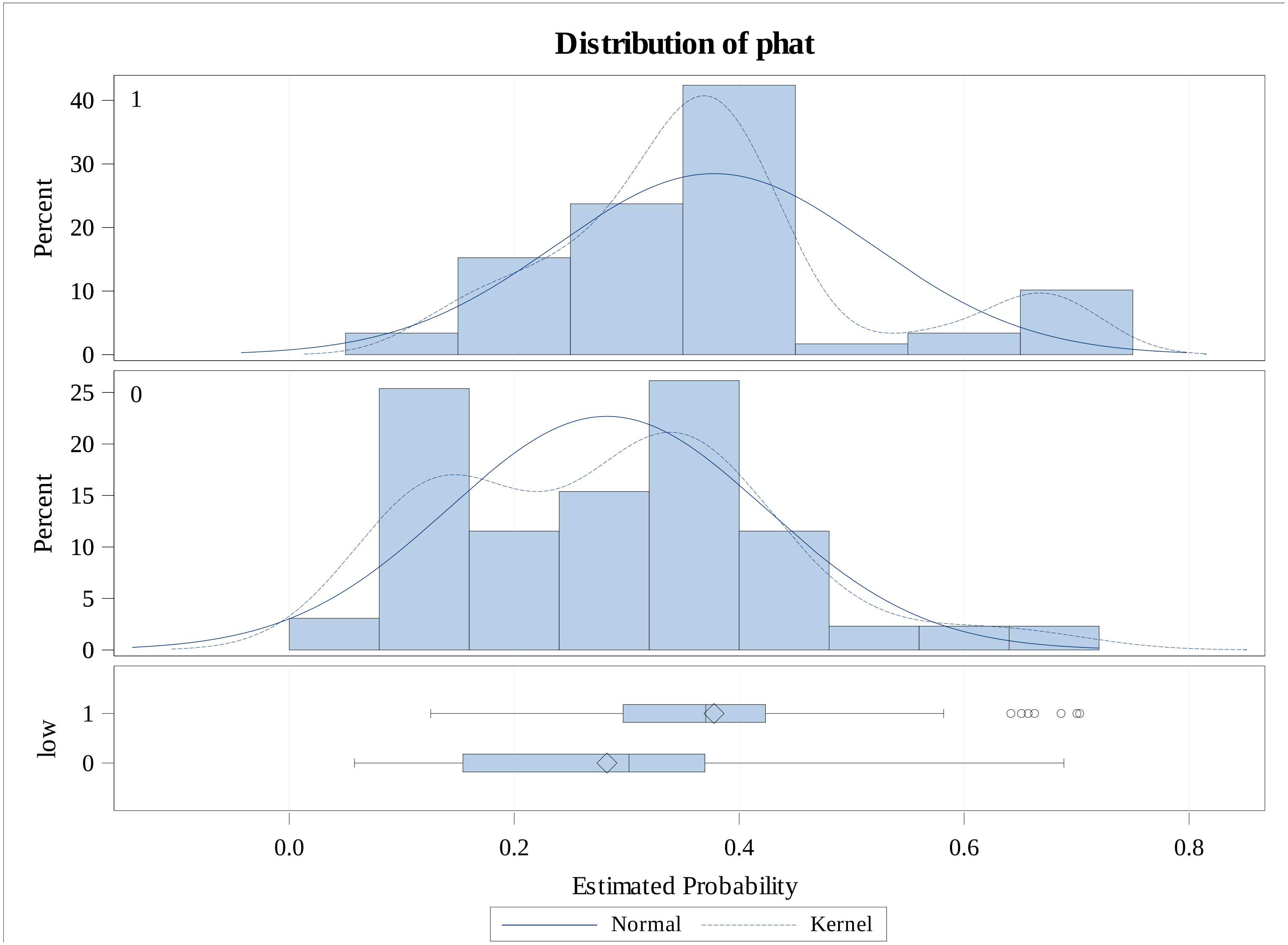
Discrimination Slope – SAS

Coefficient of Discrimination and Plots
The TTEST Procedure

Variable: phat (Estimated Probability)

low	Method	N	Mean	Std Dev	Std Err	Minimum	Maximum
1		59	0.3776	0.1401	0.0182	0.1258	0.7028
0		130	0.2825	0.1407	0.0123	0.0580	0.6887
Diff (1-2)	Pooled		0.0952	0.1405	0.0221		
Diff (1-2)	Satterthwaite		0.0952		0.0220		

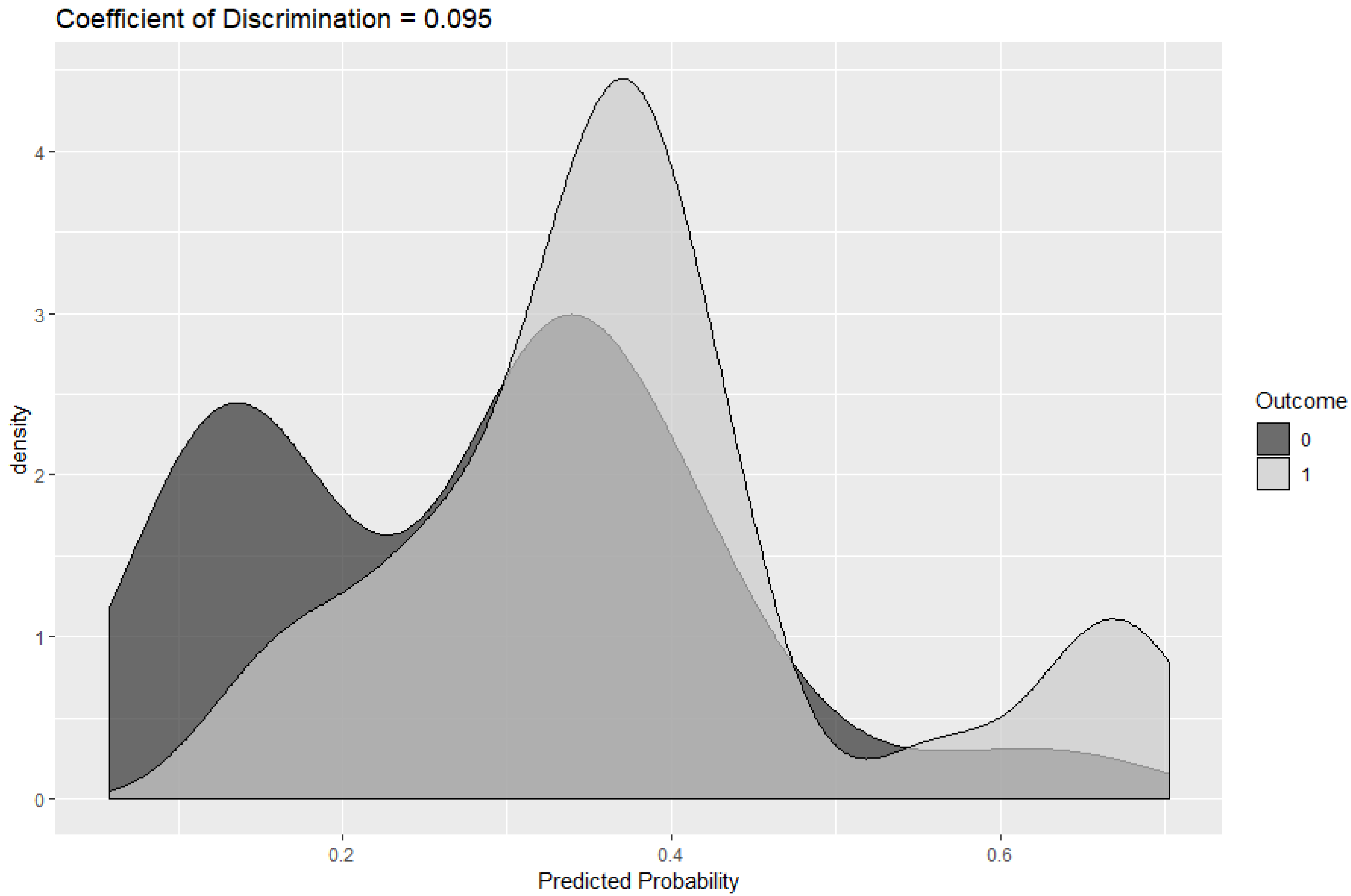
Discrimination Slope – SAS



Discrimination Slope – R

```r
bwt$p_hat <- predict(logit.model, type = "response")

p1 <- bwt$p_hat[bwt$low == 1]
p0 <- bwt$p_hat[bwt$low == 0]
coef_discrim <- mean(p1) - mean(p0)

ggplot(bwt, aes(p_hat, fill = factor(low))) +
  geom_density(alpha = 0.7) +
  scale_fill_grey() +
  labs(x = "Predicted Probability",
       fill = "Outcome",
       title = paste("Coefficient of Discrimination = ",
                     round(coef_discrim, 3), sep = ""))
```

Discrimination Slope – R



Rank-order Statistics

- How well does the model order predictions?

- **Concordance:** for a pair of subjects with and without the event, the one **with the event** had the **higher** predicted probability.
- **Discordance:** for a pair of subjects with and without the event, the one **with the event** had the **lower** predicted probability.
- **Tied:** for a pair of subjects with and without the event, they both have the **same** predicted probability.

Concordance

- **Interpretation** – For all possible (1,0) pairs, the model assigned the higher predicted probability to the observation with the event *concordance%* of the time.
- Common metrics based on concordance:
  - c-statistic: $c = Concordance\,\% + \frac{1}{2} Tied\,\%$
  - Somers' D: $D_{xy} = 2c - 1$
  - Kendall's τ_a: $\tau_a = \dfrac{\#\text{concordant} - \#\text{discordant}}{\frac{n(n-1)}{2}}$

Rank-order Statistics – SAS

```
proc logistic data=logistic.lowbwt plots(only)=(oddsratio);
   class race(ref='white') / param=ref;
   model low(event='1') = race lwt smoke / clodds=pl
                                            clparm=pl;
   title 'Modeling Low Birth Weight';
run;
quit;
```

Rank-order Statistics – SAS

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	68.3	**Somers' D**	0.371
Percent Discordant	31.2	**Gamma**	0.373
Percent Tied	0.5	**Tau-a**	0.160
Pairs	7670	**c**	0.686

Rank-order Statistics – R

```
Concordance(bwt$low, bwt$p_hat)
```

```
## $Concordance
## [1] 0.6831812
##
## $Discordance
## [1] 0.3168188
##
## $Tied
## [1] 5.551115e-17
##
## $Pairs
## [1] 7670
```

```
somersD(bwt$low, bwt$p_hat)
```

```
## [1] 0.3663625
```

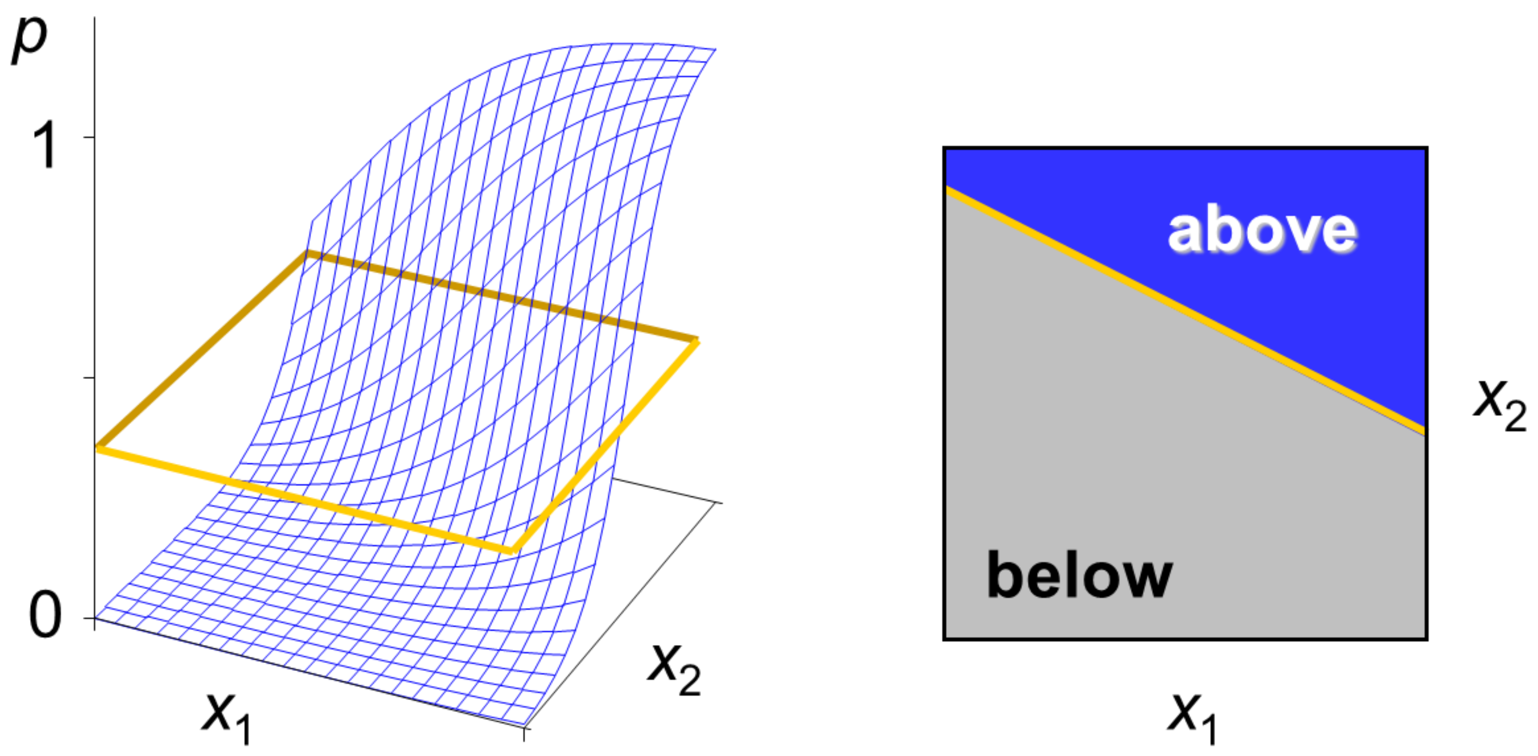
ASSESSING PREDICTIVE POWER

Classification Based Metrics

Classification

- Want model to correctly classify events and non-events.
- **Classification** forces the model to predict $\hat{y}_i = 1$ or $\hat{y}_i = 0$ based on whether the predicted probability exceeds some threshold – for example, $\hat{y}_i = 1$ if $\hat{p}_i > 0.5$.
- Strict classification-based measures completely discard any information about the actual quality of the model's predicted probabilities.

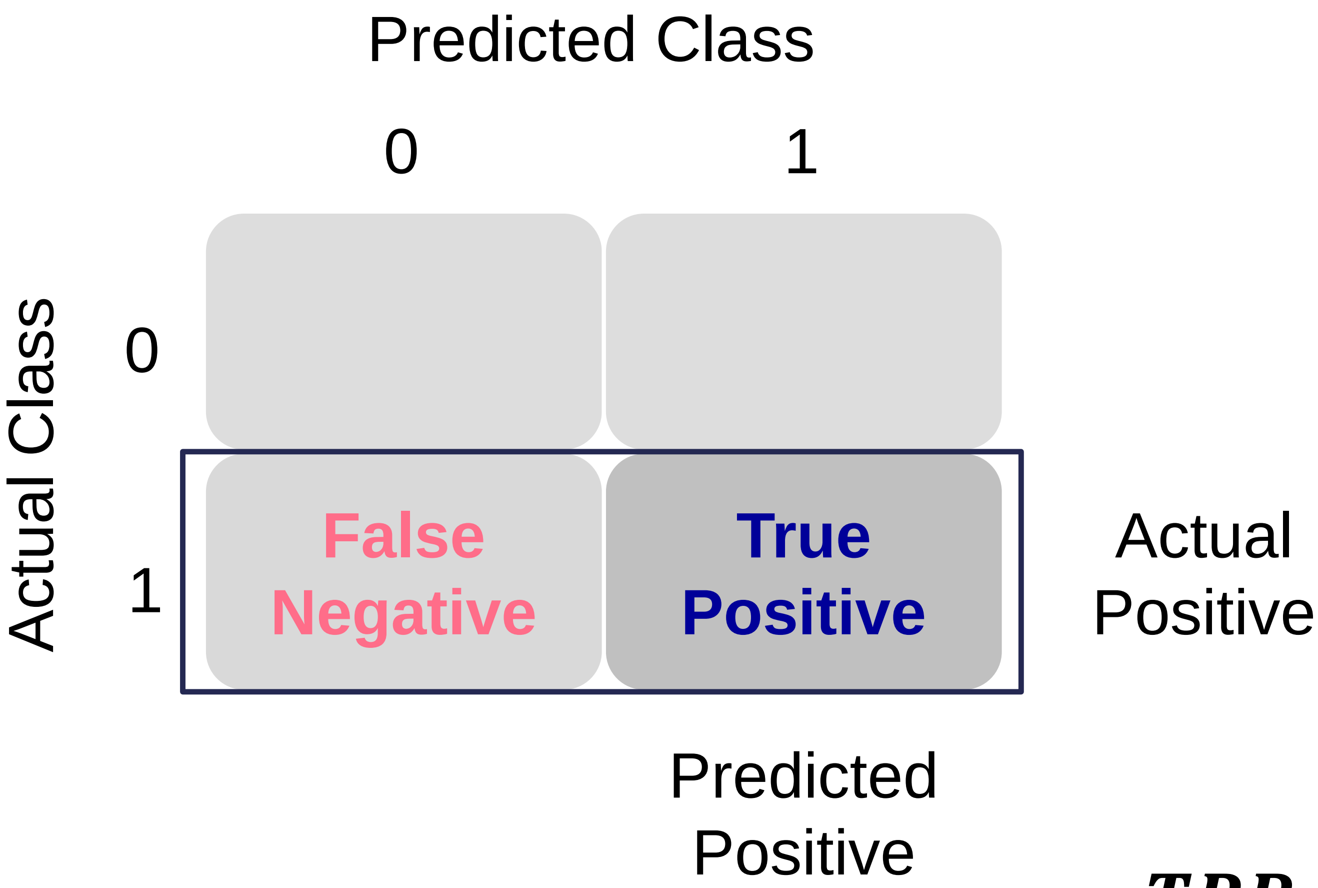
Logistic Discrimination



Classification Table

		Predicted Class 0	Predicted Class 1	
Actual Class	0	**True Negative**	**False Positive**	Actual Negative
	1	**False Negative**	**True Positive**	Actual Positive
		Predicted Negative	Predicted Positive	

Sensitivity / Recall



$$TPR = \frac{TP}{TP + FN}$$

Specificity

		Predicted Class		
		0	1	
Actual Class	0	**True Negative**	**False Positive**	Actual Negative
	1			
		Predicted Negative		

$$\boldsymbol{TNR = \frac{TN}{TN + FP}}$$

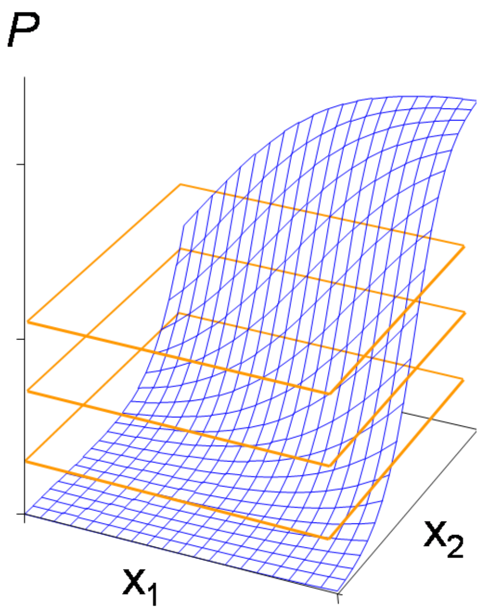
1 – Specificity

Predicted Class

Actual Class		0	1	
	0	**True Negative**	**False Positive**	Actual Negative
	1			
		Predicted Negative		

$$\boldsymbol{FPR = \frac{FP}{TN + FP}}$$

Classification Changes with Cut-off



response	$\hat{P}$	cutoff=.5	cutoff=.25
0	.32	0	1
1	.40	0	1
1	.92	1	1
0	.06	0	0
1	.52	1	1
1	.39	0	1
1	.22	0	0
0	.17	0	0
0	.13	0	0
⋮	⋮	⋮	⋮
1	.75	1	1

Best Cut-off?

- **Always** consider the cost of false positives and false negatives when doing classification.
- When **NOT** considering costs, many different techniques to “optimal” cut-off.
- **Youden J statistic** (or **Youden’s index**):

$$J = \text{sensitivity} + \text{specificity} - 1$$

- “Optimal” – false positives and false negatives are weighed equally , so select cut-off that produces highest Youden *J* statistic.

Classification Table – SAS

```
proc logistic data=logistic.lowbwt plots(only)=(oddsratio);
   class race(ref='white') / param=ref;
   model low(event='1') = race lwt smoke /
                       ctable pprob = 0 to 0.98 by 0.02;
   ods output classification=classtable;
   title 'Modeling Low Birth Weight';
run;
quit;
```

Classification Table – SAS

Classification Table									
Prob Level	Correct		Incorrect		Percentages				
	Event	Non-Event	Event	Non-Event	Correct	Sensi-tivity	Speci-ficity	Pos Pred	Neg Pred
0.000	59	0	130	0	31.2	100.0	0.0	31.2	.
0.020	59	0	130	0	31.2	100.0	0.0	31.2	.
0.040	59	0	130	0	31.2	100.0	0.0	31.2	.
0.060	59	1	129	0	31.7	100.0	0.8	31.4	100.0
0.080	59	4	126	0	33.3	100.0	3.1	31.9	100.0
0.100	59	10	120	0	36.5	100.0	7.7	33.0	100.0
0.120	58	15	115	1	38.6	98.3	11.5	33.5	93.8
0.140	57	26	104	2	43.9	96.6	20.0	35.4	92.9
0.160	54	35	95	5	47.1	91.5	26.9	36.2	87.5
0.180	53	43	87	6	50.8	89.8	33.1	37.9	87.8

⋮

Youden's Index – SAS

```
data classtable;
   set classtable;
   youden = sensitivity + specificity - 100;
   drop PPV NPV Correct;
run;

proc sort data=classtable;
      by descending youden;
run;

proc print data=classtable;
run;
```

Youden's Index – SAS

Obs	ProbLevel	Sensitivity	Specificity	FalsePositive	FalseNegative	youden
1	0.200	89.8	34.6	61.6	11.8	24.4459
2	0.180	89.8	33.1	62.1	12.2	22.9074
3	0.220	86.4	36.2	61.9	14.5	22.5945
4	0.240	81.4	39.2	62.2	17.7	20.5867
5	0.300	72.9	47.7	61.3	20.5	20.5737

⋮

Classification Table – R

```r
confusionMatrix(bwt$low, bwt$p_hat, threshold = 0.5)
```

```
##     0  1
## 0 122 50
## 1   8  9
```

Classification & Youden Index – R

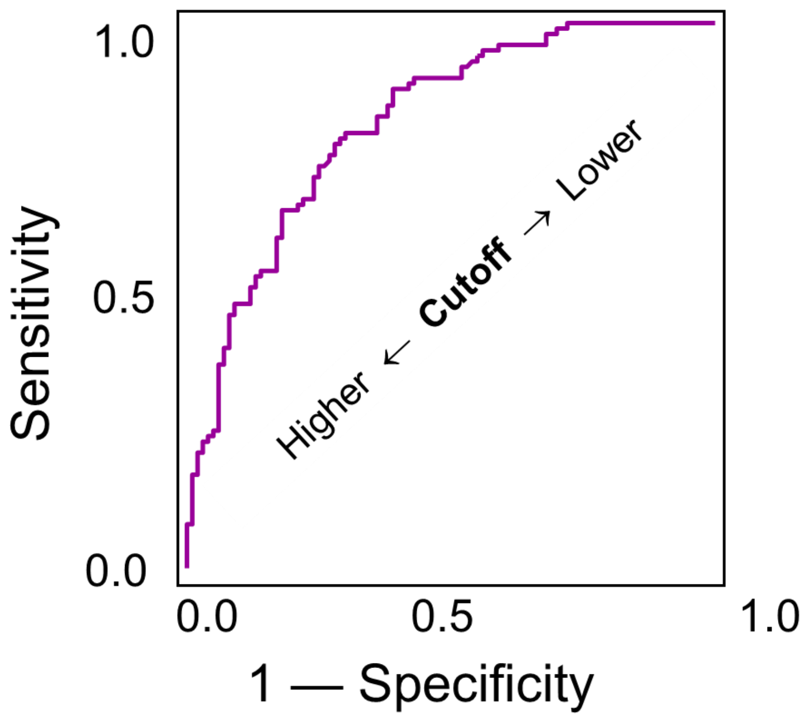
```
sens <- NULL
spec <- NULL
youden <- NULL
cutoff <- NULL

for(i in 1:49){
  cutoff = c(cutoff, i/50)
  sens <- c(sens, sensitivity(bwt$low, bwt$p_hat,
              threshold = i/50))
  spec <- c(spec, specificity(bwt$low, bwt$p_hat,
              threshold = i/50))
  youden <- c(youden, youdensIndex(bwt$low, bwt$p_hat,
                threshold = i/50))
}

ctable <- data.frame(cutoff, sens, spec, youden)
```

Classification & Youden Index – R

```
##    cutoff       sens        spec      youden
## 1    0.02 1.00000000 0.000000000 0.000000000
## 2    0.04 1.00000000 0.000000000 0.000000000
## 3    0.06 1.00000000 0.007692308 0.007692308
## 4    0.08 1.00000000 0.030769231 0.030769231
## 5    0.10 1.00000000 0.084615385 0.084615385
## 6    0.12 1.00000000 0.123076923 0.123076923
## 7    0.14 0.96610169 0.230769231 0.196870926
## 8    0.16 0.96610169 0.284615385 0.250717080
## 9    0.18 0.93220339 0.330769231 0.262972621
## 10   0.20 0.89830508 0.353846154 0.252151239
## 11   0.22 0.89830508 0.369230769 0.267535854
## 12   0.24 0.86440678 0.400000000 0.264406780
## 13   0.26 0.81355932 0.415384615 0.228943937
```

⋮

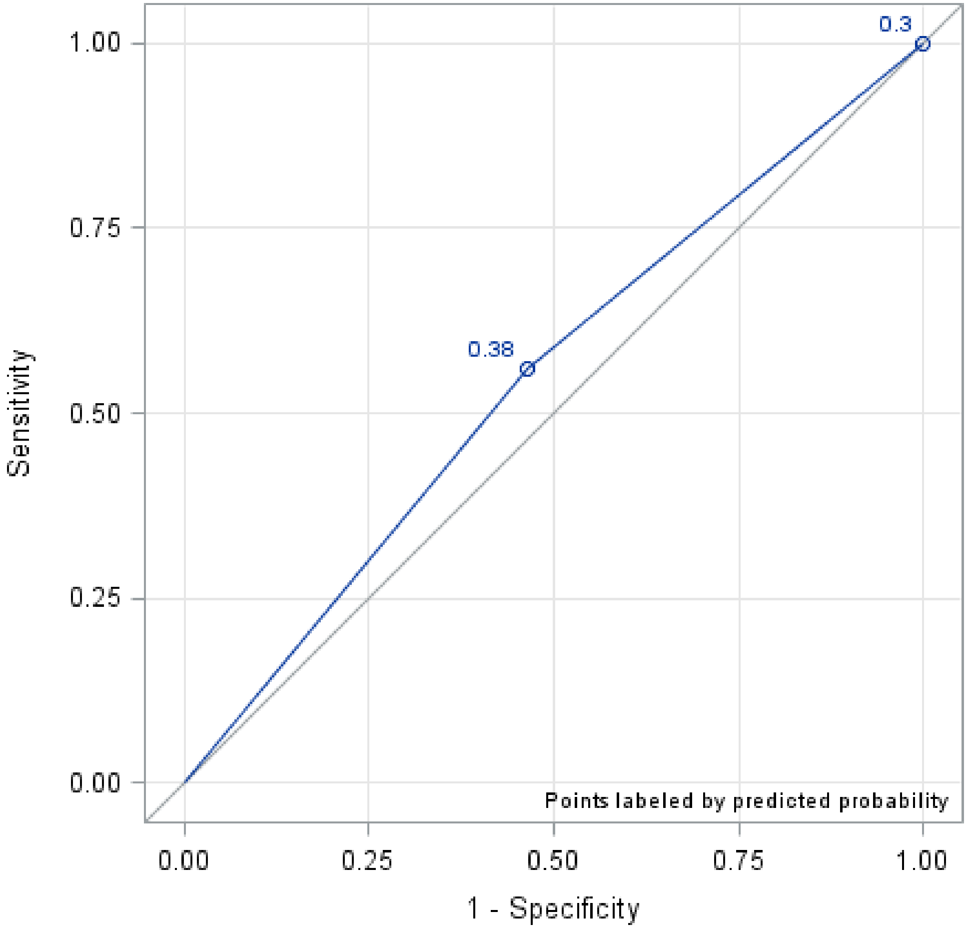
ROC Curve



- **ROC curve** plots *TPR* vs. *FPR* for a grid of thresholds.
- **Area under the curve** (AUC or AUROC) summarizes the overall quality of ROC curve – equivalent to c-statistic.
- Want high sensitivity and high specificity.

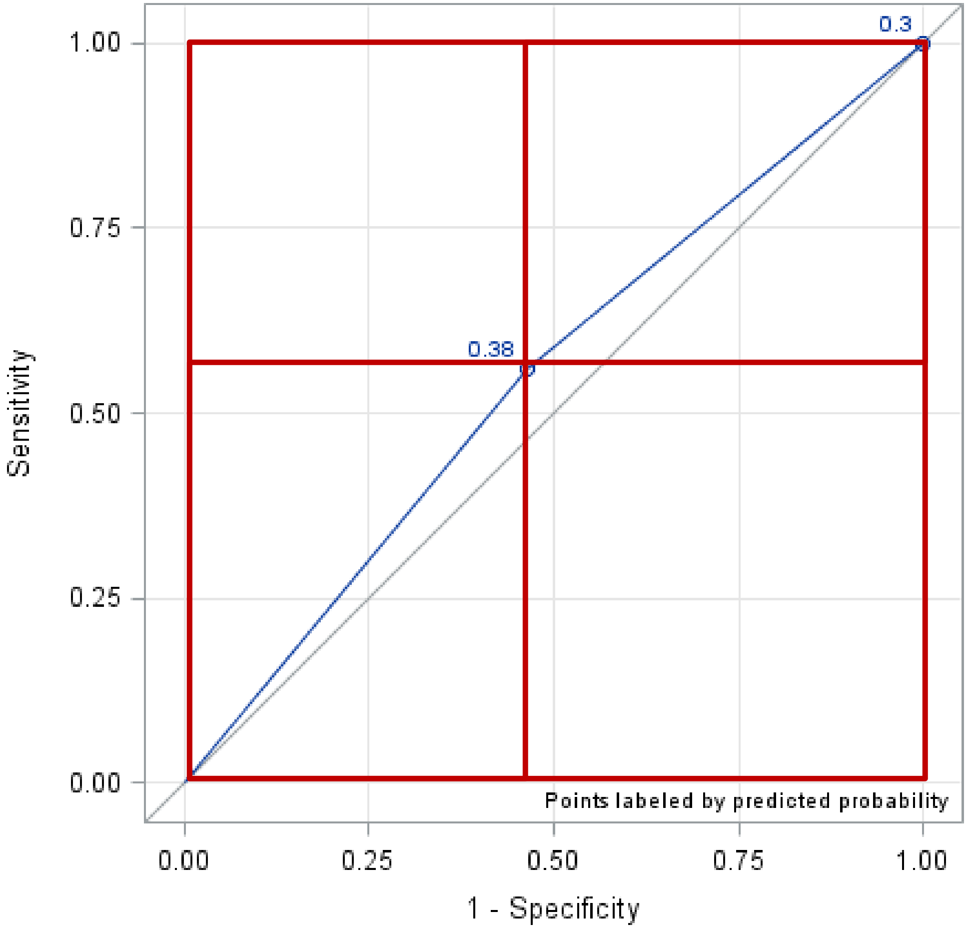
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



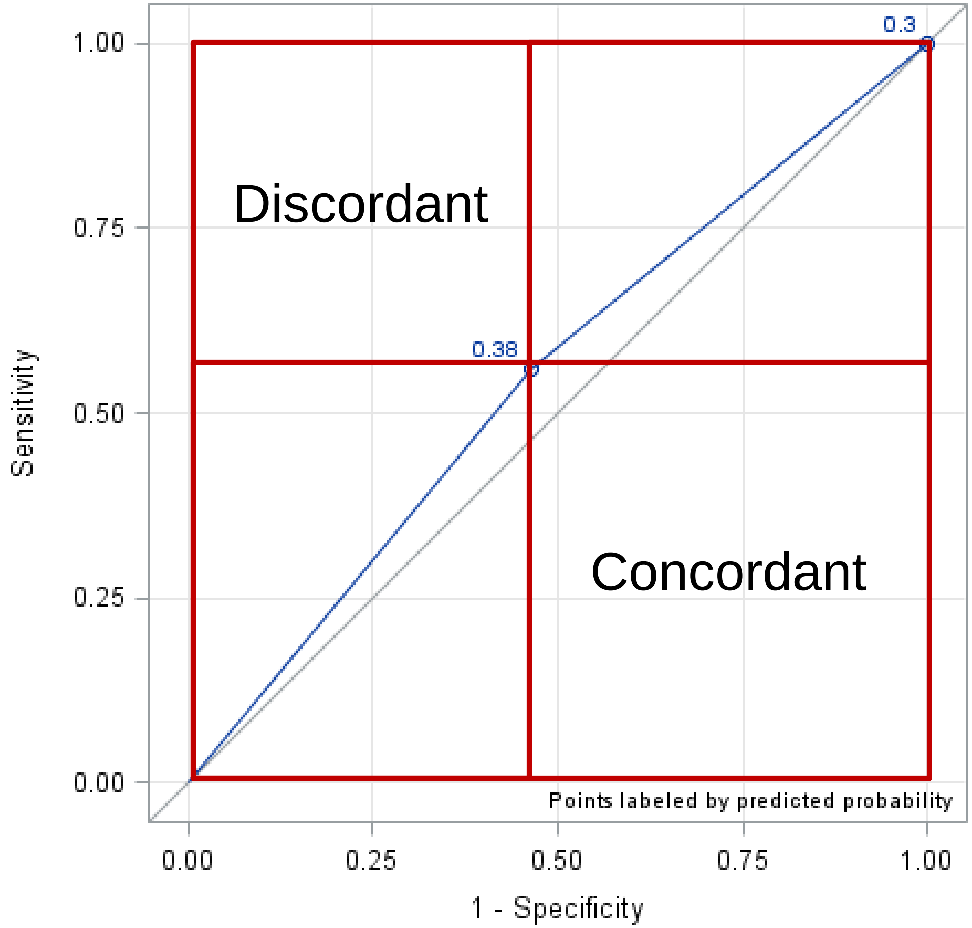
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



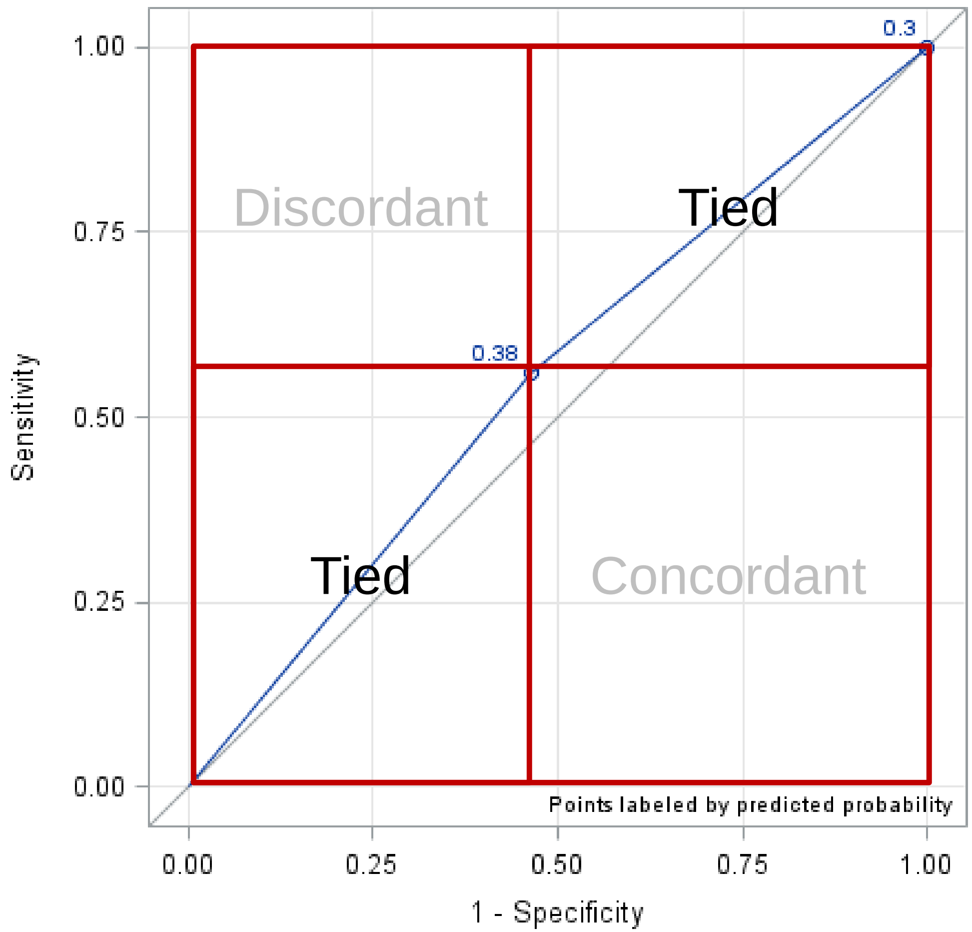
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



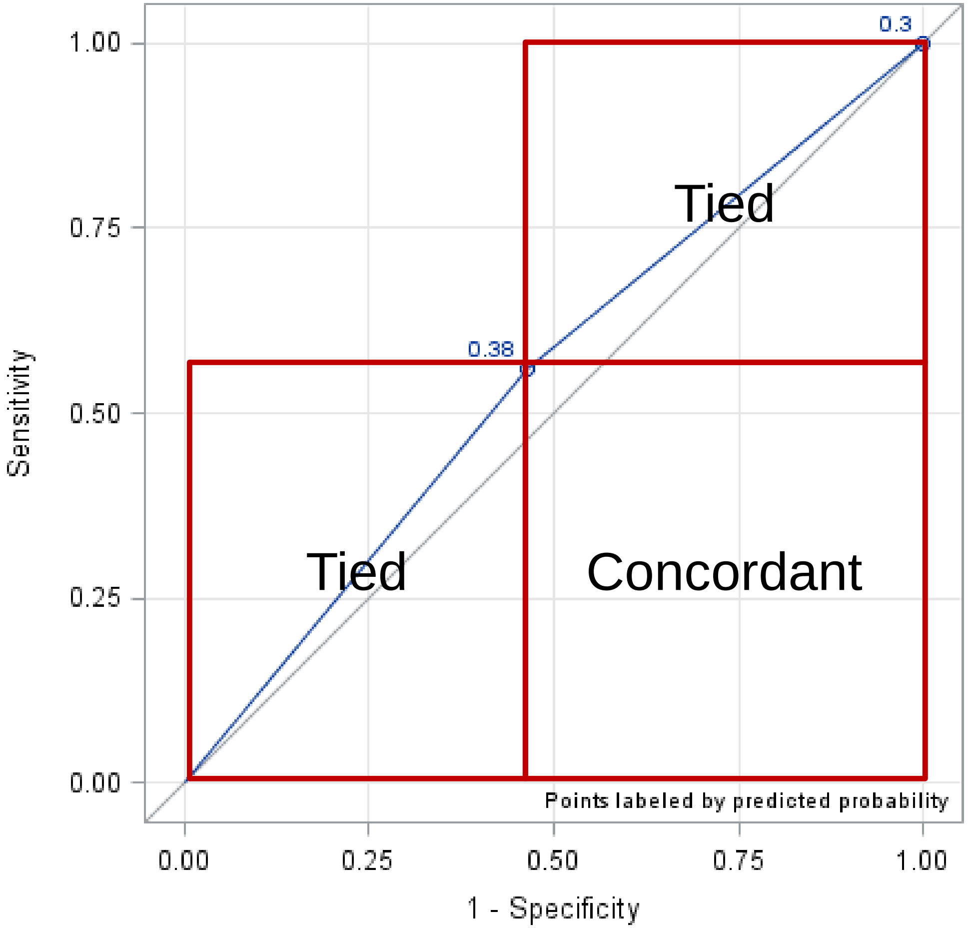
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



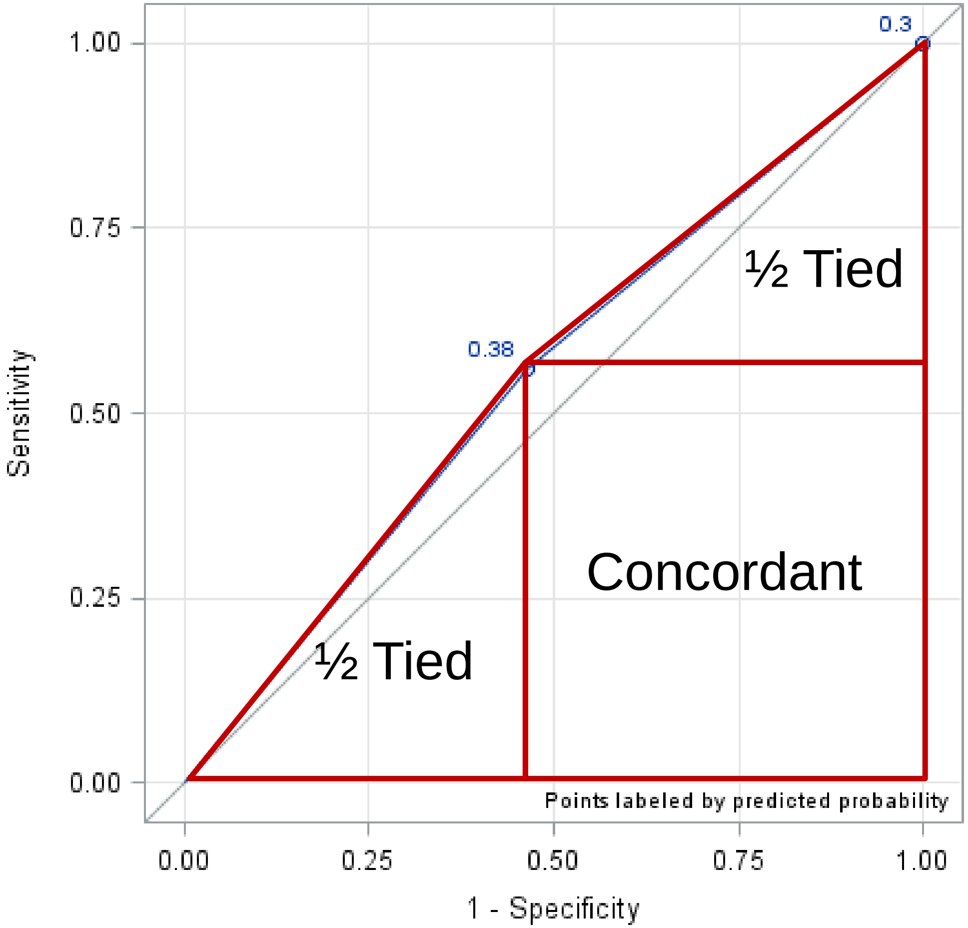
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



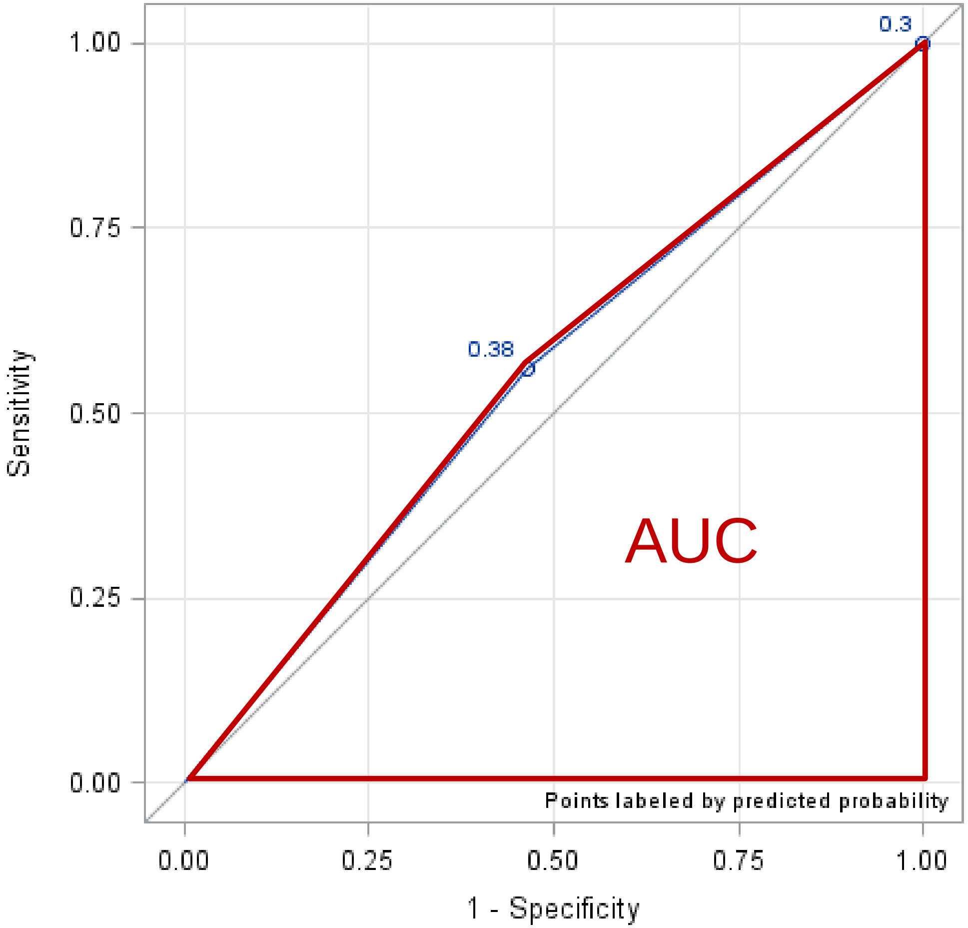
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



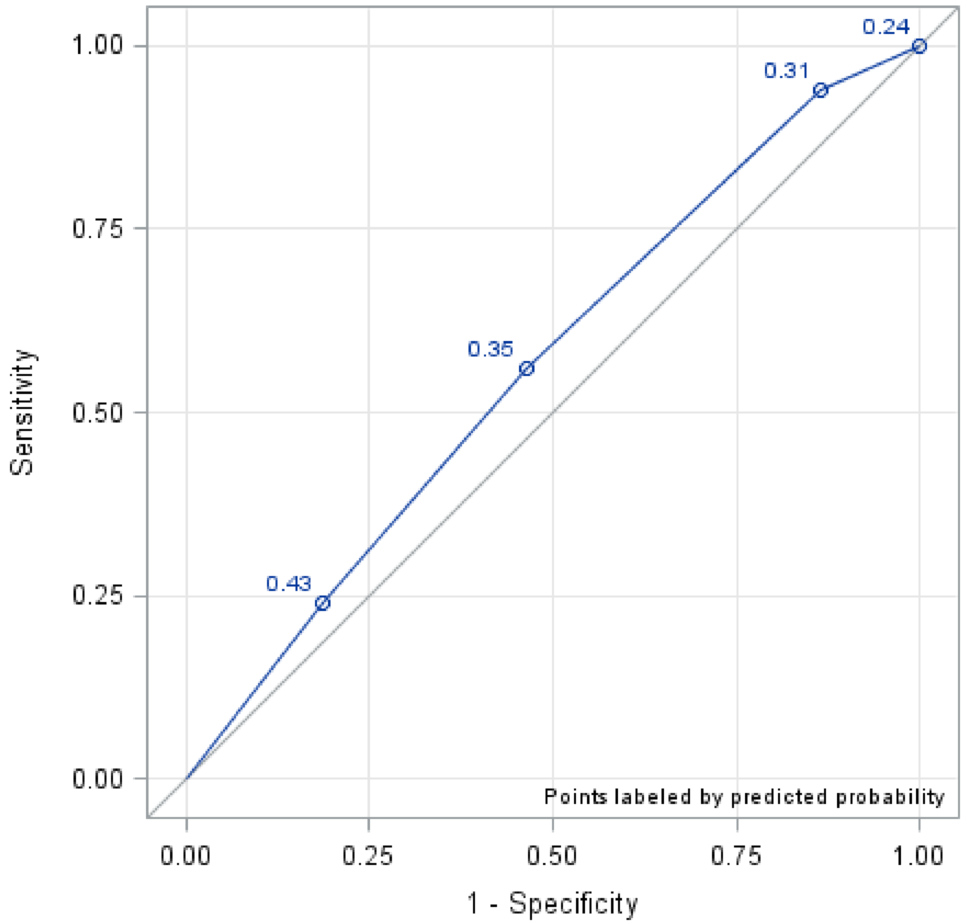
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



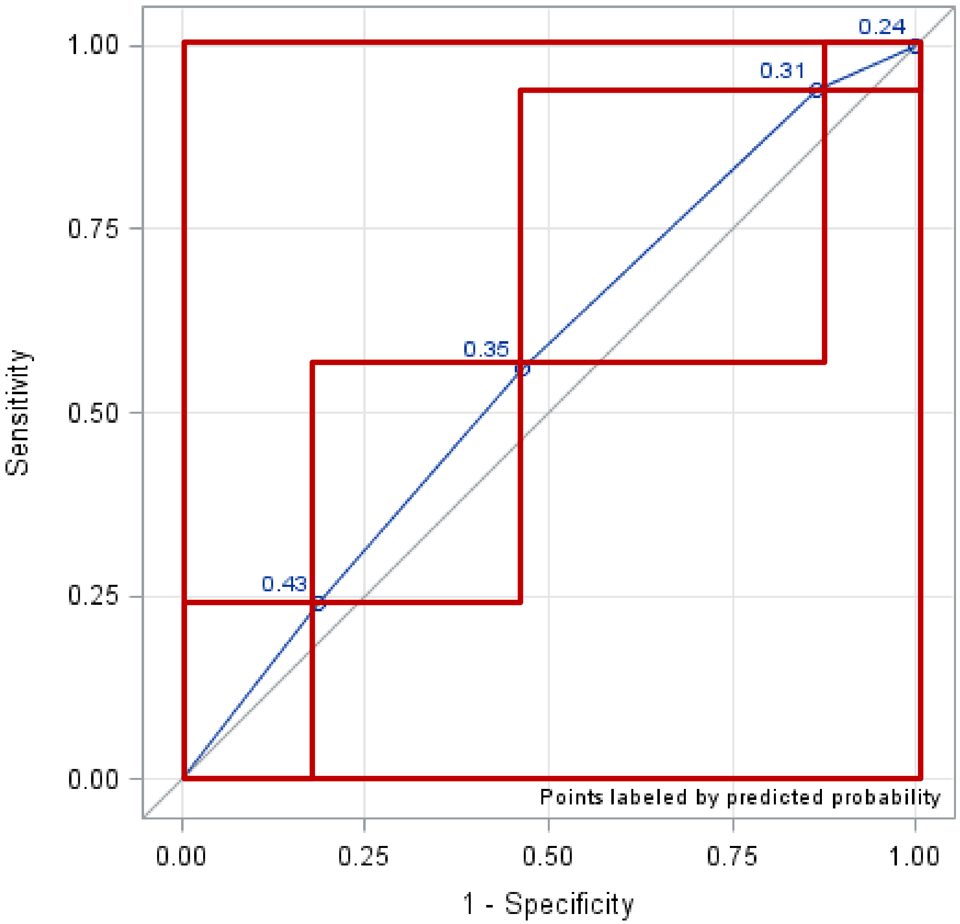
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



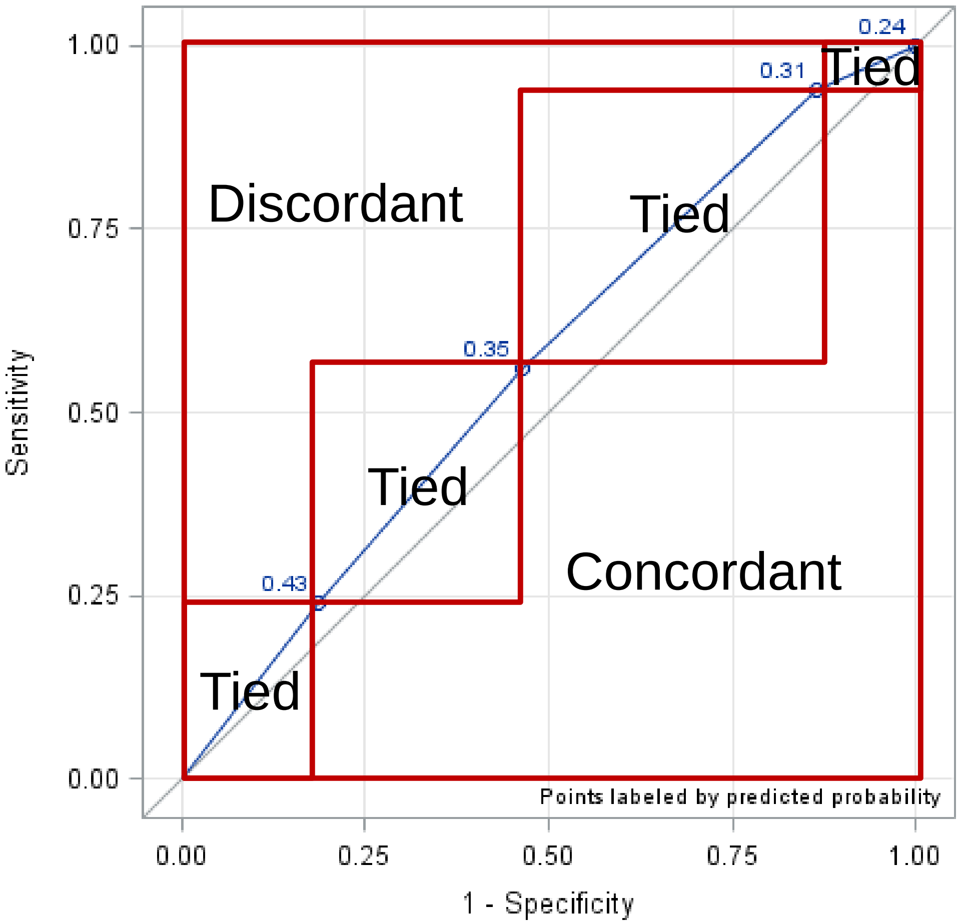
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



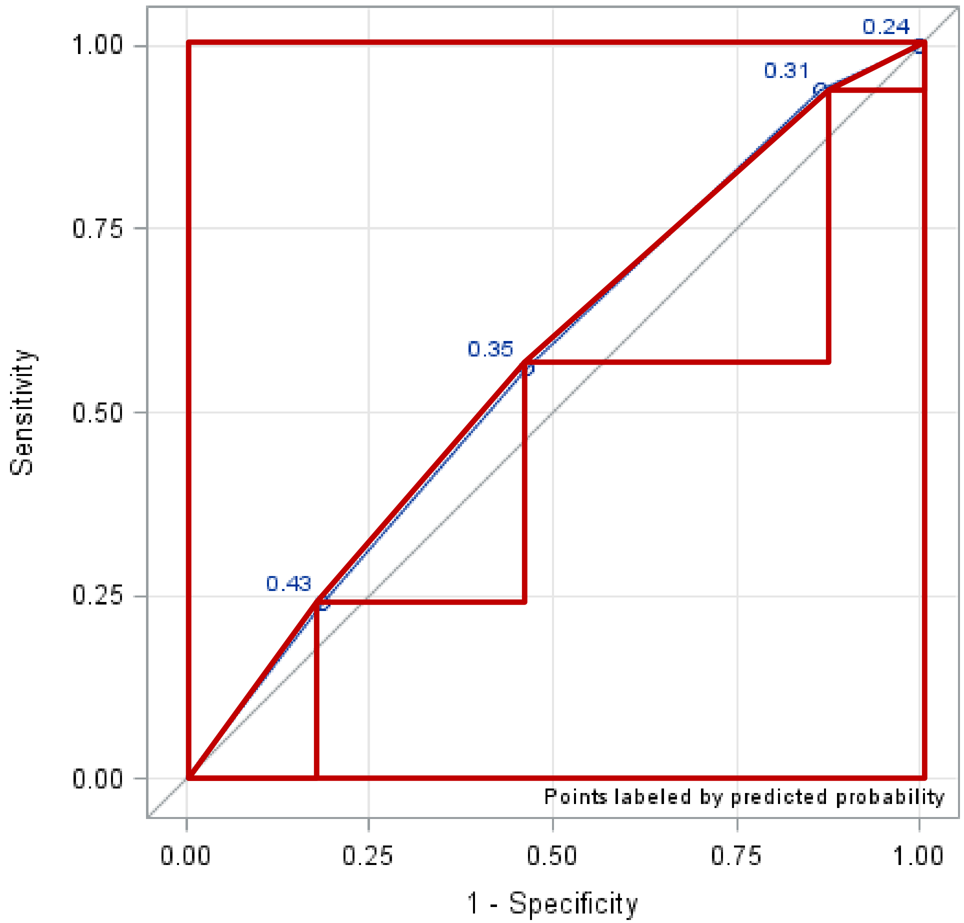
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



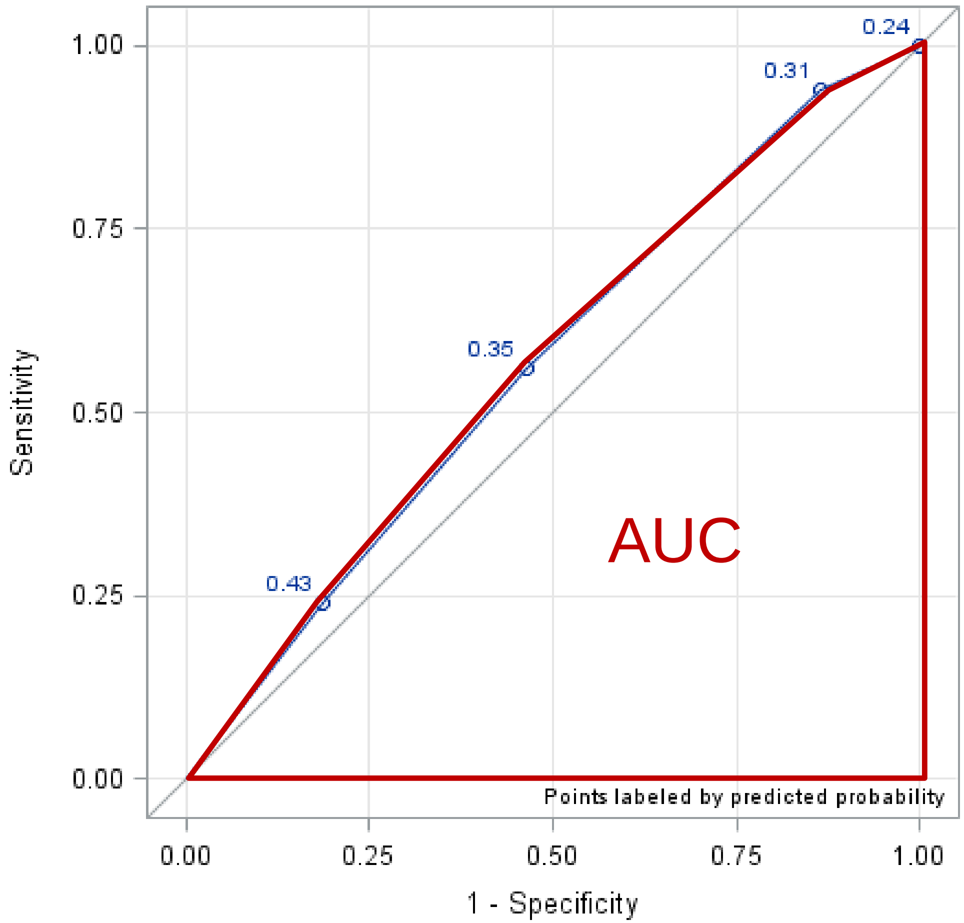
Area Under the ROC Curve

$$AUC = \%\,Concordant + \frac{1}{2}(\%\,Tied)$$



Area Under the ROC Curve

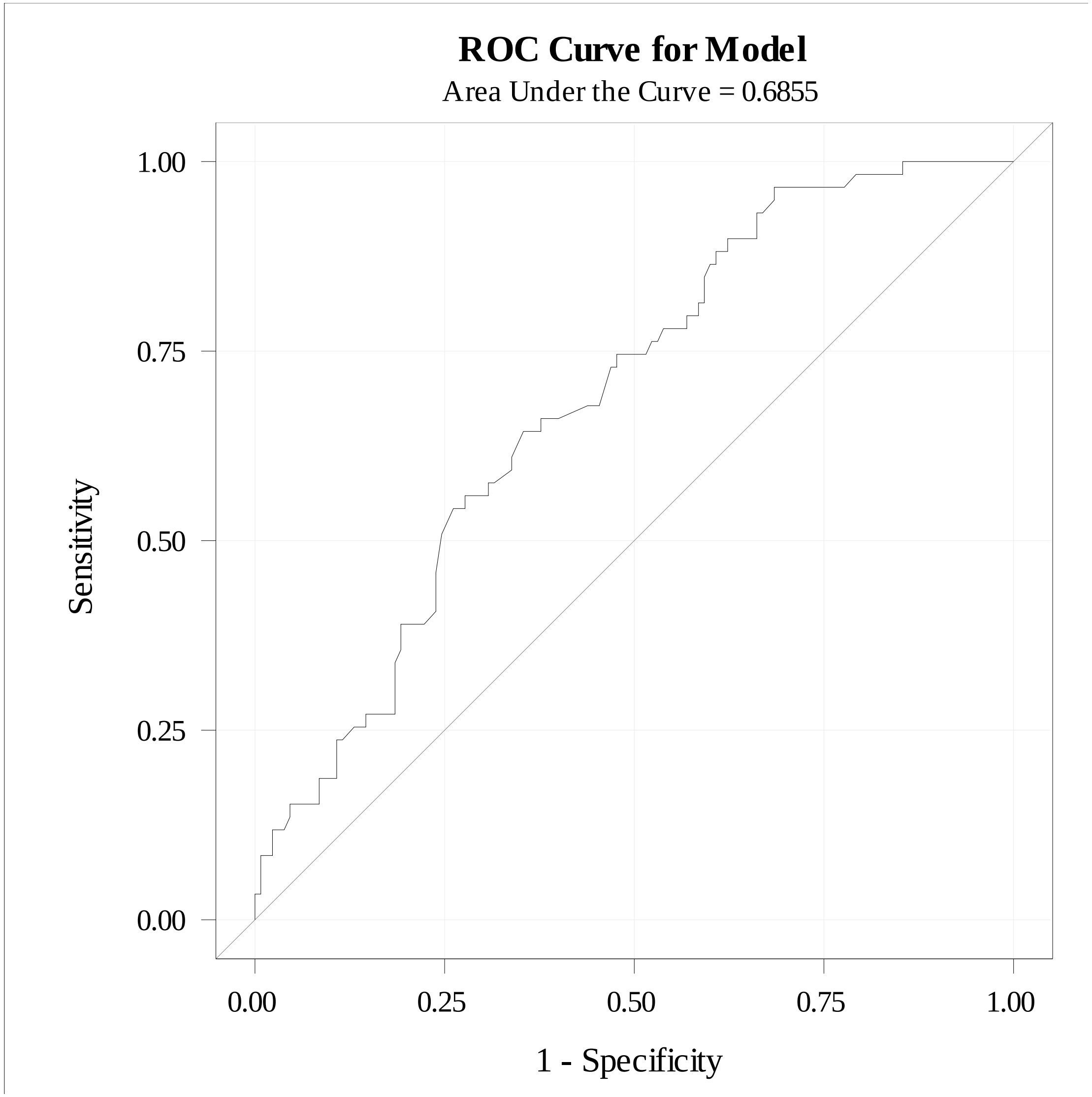
$$AUC = \%\ Concordant + \frac{1}{2}(\%\ Tied)$$



ROC Curve – SAS

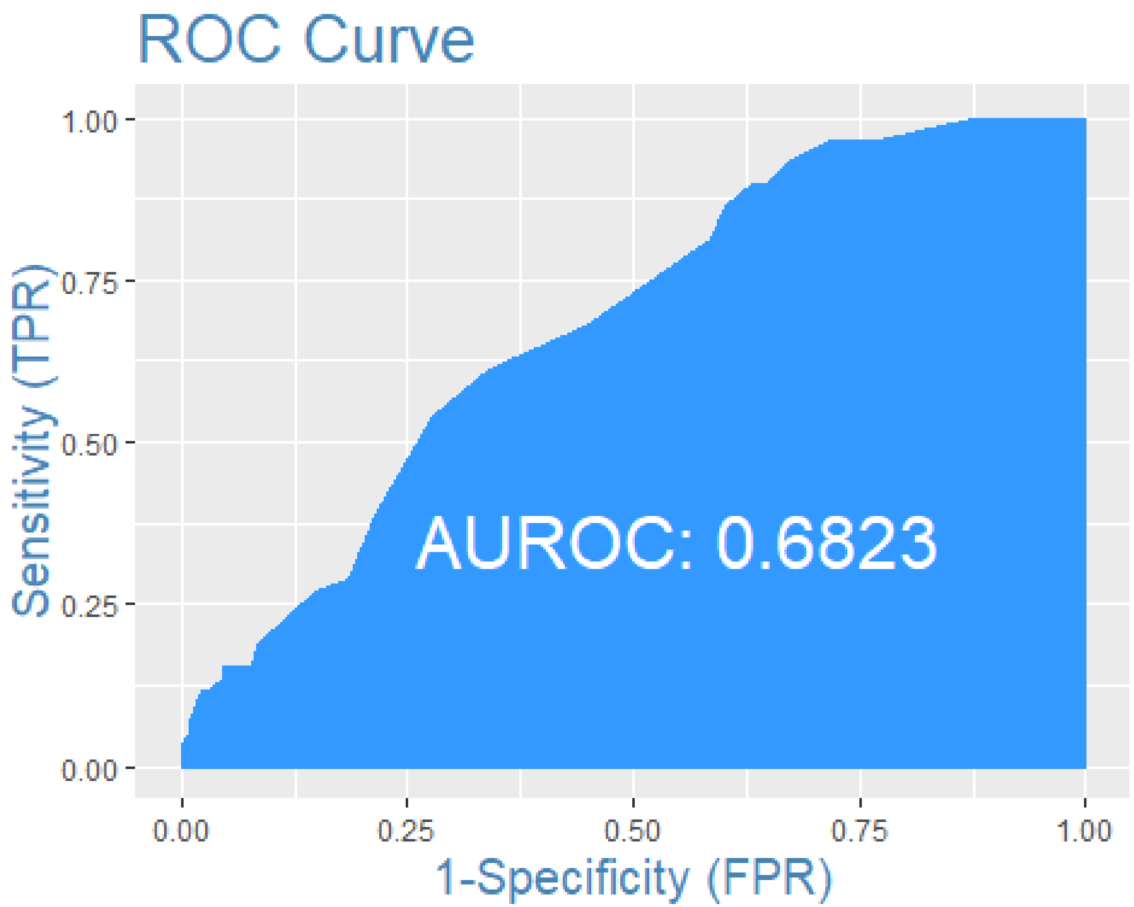
```
proc logistic data=logistic.lowbwt plots(only)=ROC;
   class race(ref='white') / param=ref;
   model low(event='1') = race lwt smoke / clodds=pl
                                            clparm=pl;
   title 'Modeling Low Birth Weight';
run;
quit;
```

ROC Curve – SAS



ROC Curve – R

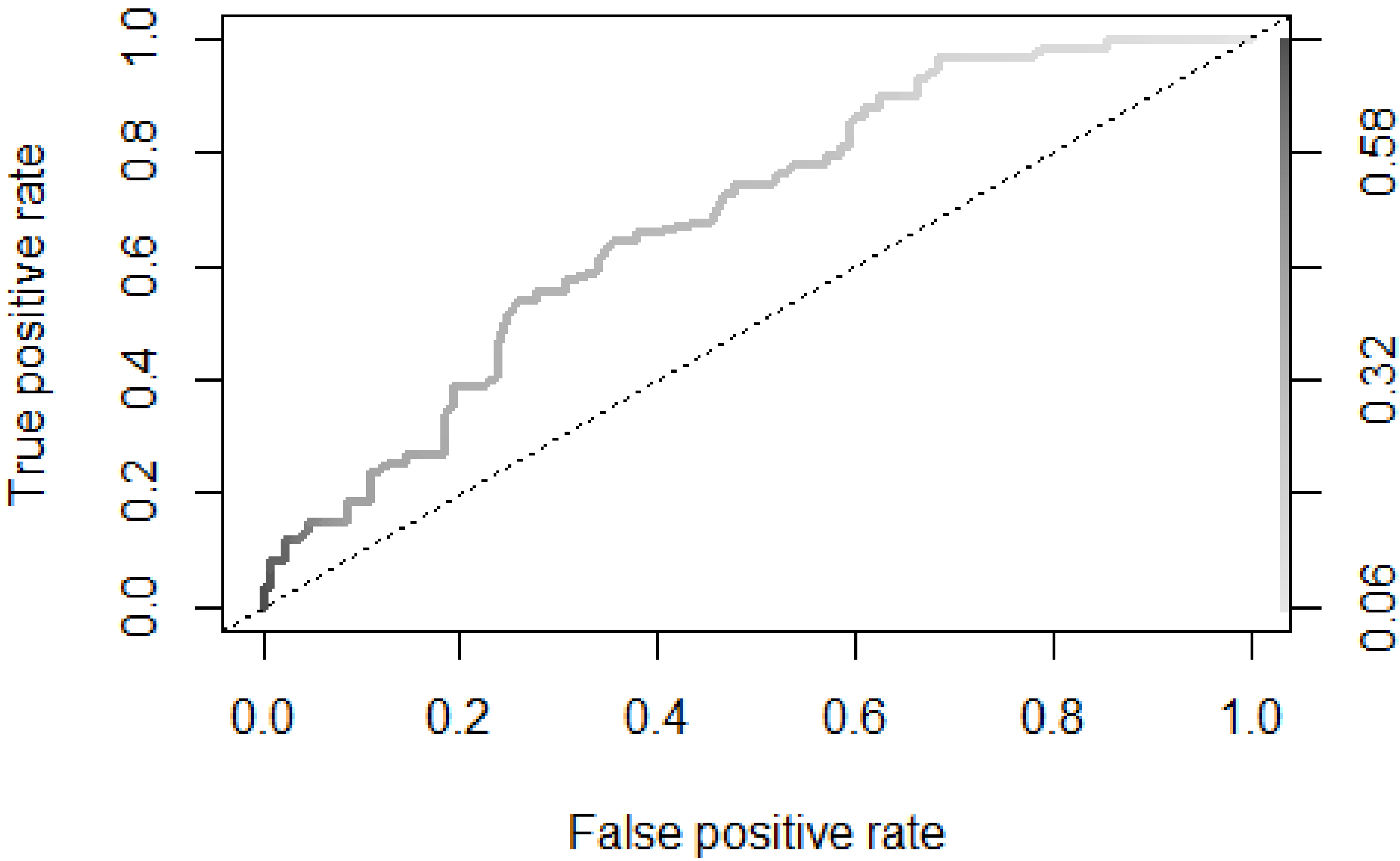
```
plotROC(bwt$low, bwt$p_hat)
```



ROC Curve – R

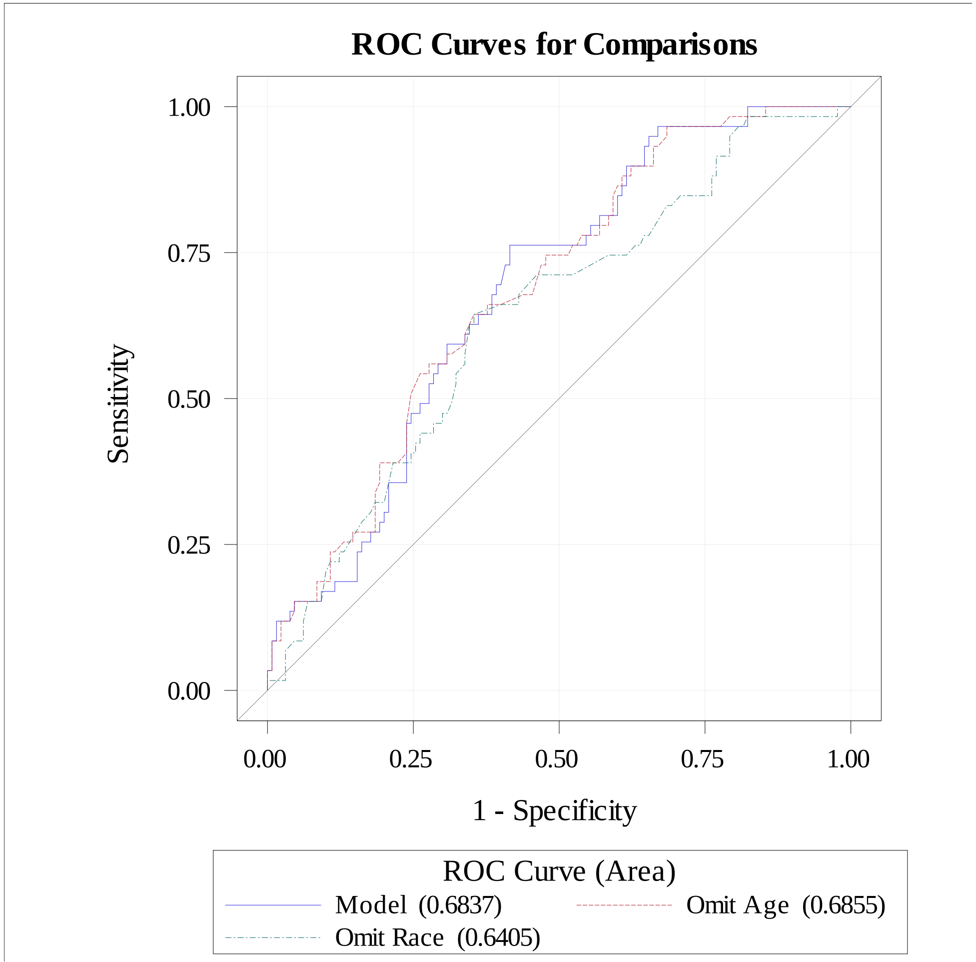
```
pred <- prediction(fitted(logit.model), factor(bwt$low))
perf <- performance(pred, measure = "tpr", x.measure = "fpr")
plot(perf, lwd = 3, colorize = TRUE, colorkey = TRUE,
     colorize.palette = rev(gray.colors(256)))
abline(a = 0, b = 1, lty = 3)
```

ROC Curve – R


True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
0.06
0.32
0.58

Comparing ROC Curves

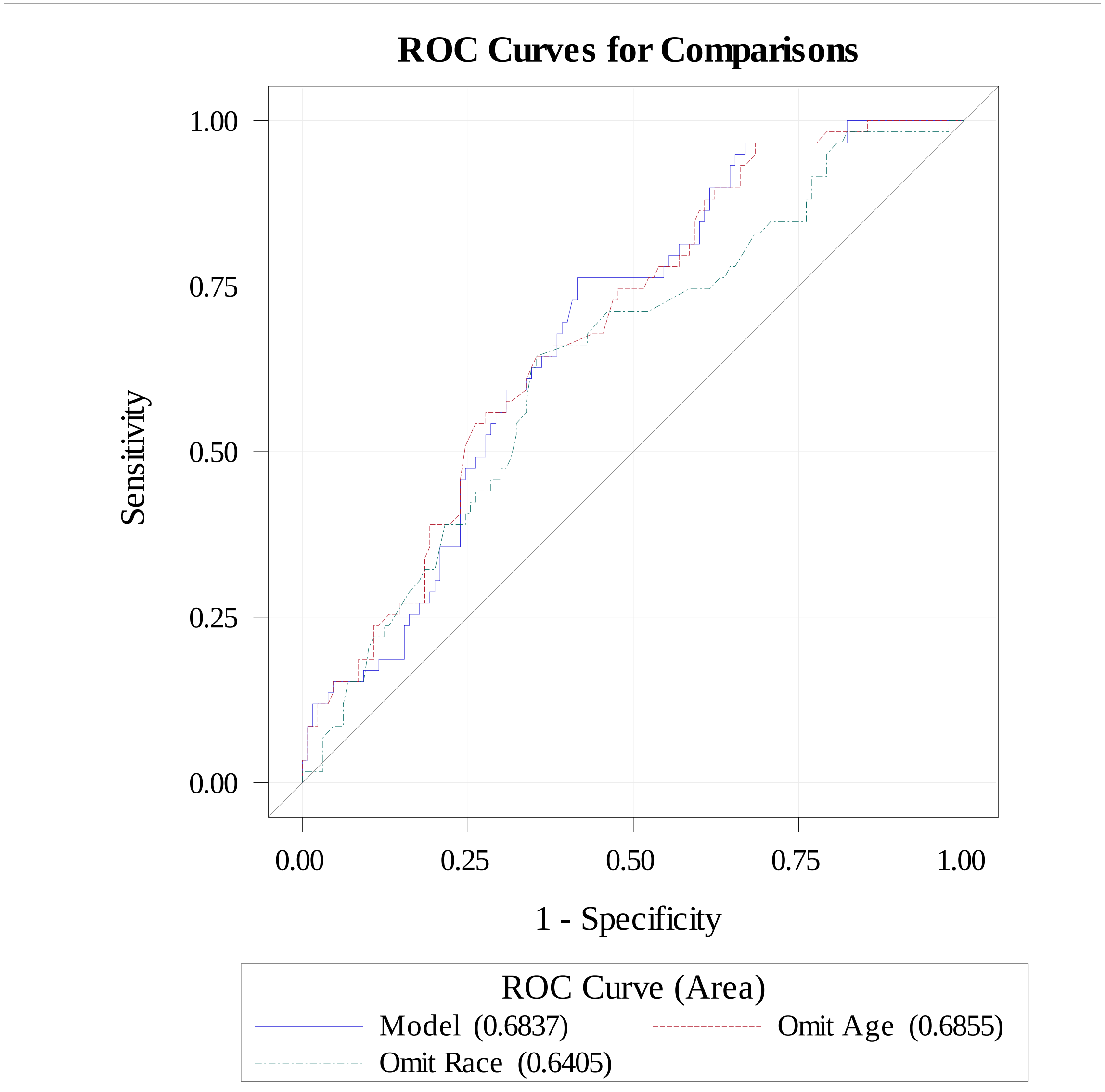
- Can compare different ROC curves statistically by their AUC.



Comparing ROC Curves – SAS

```
proc logistic data=logistic.lowbwt plots(only)=ROC;
   class race(ref='white') / param=ref;
   model low(event='1') = age race lwt smoke / clodds=pl
                                                clparm=pl;
   ROC 'Omit Age' race lwt smoke;
   ROC 'Omit Race' lwt smoke;
   ROCcontrast / estimate = allpairs;
   title 'Comparing ROC Curves';
run;
quit;
```

Comparing ROC Curves – SAS



Comparing ROC Curves – SAS

ROC Association Statistics							
ROC Model	Mann-Whitney				Somers' D	Gamma	Tau-a
	Area	Standard Error	95% Wald Confidence Limits				
Model	0.6837	0.0393	0.6068	0.7606	0.3674	0.3675	0.1586
Omit Age	0.6855	0.0395	0.6081	0.7630	0.3711	0.3728	0.1602
Omit Race	0.6405	0.0428	0.5567	0.7243	0.2810	0.2843	0.1213

Comparing ROC Curves – SAS

ROC Contrast Test Results			
Contrast	**DF**	**Chi-Square**	**Pr > ChiSq**
Reference = Model	2	1.7008	0.4273

ROC Contrast Estimation and Testing Results by Row						
Contrast	**Estimate**	**Standard Error**	**95% Wald Confidence Limits**		**Chi-Square**	**Pr > ChiSq**
Model - Omit Age	-0.00183	0.00940	-0.0202	0.0166	0.0377	0.8460
Model - Omit Race	0.0432	0.0344	-0.0242	0.1107	1.5772	0.2092
Omit Age - Omit Race	0.0450	0.0345	-0.0227	0.1127	1.7008	0.1922

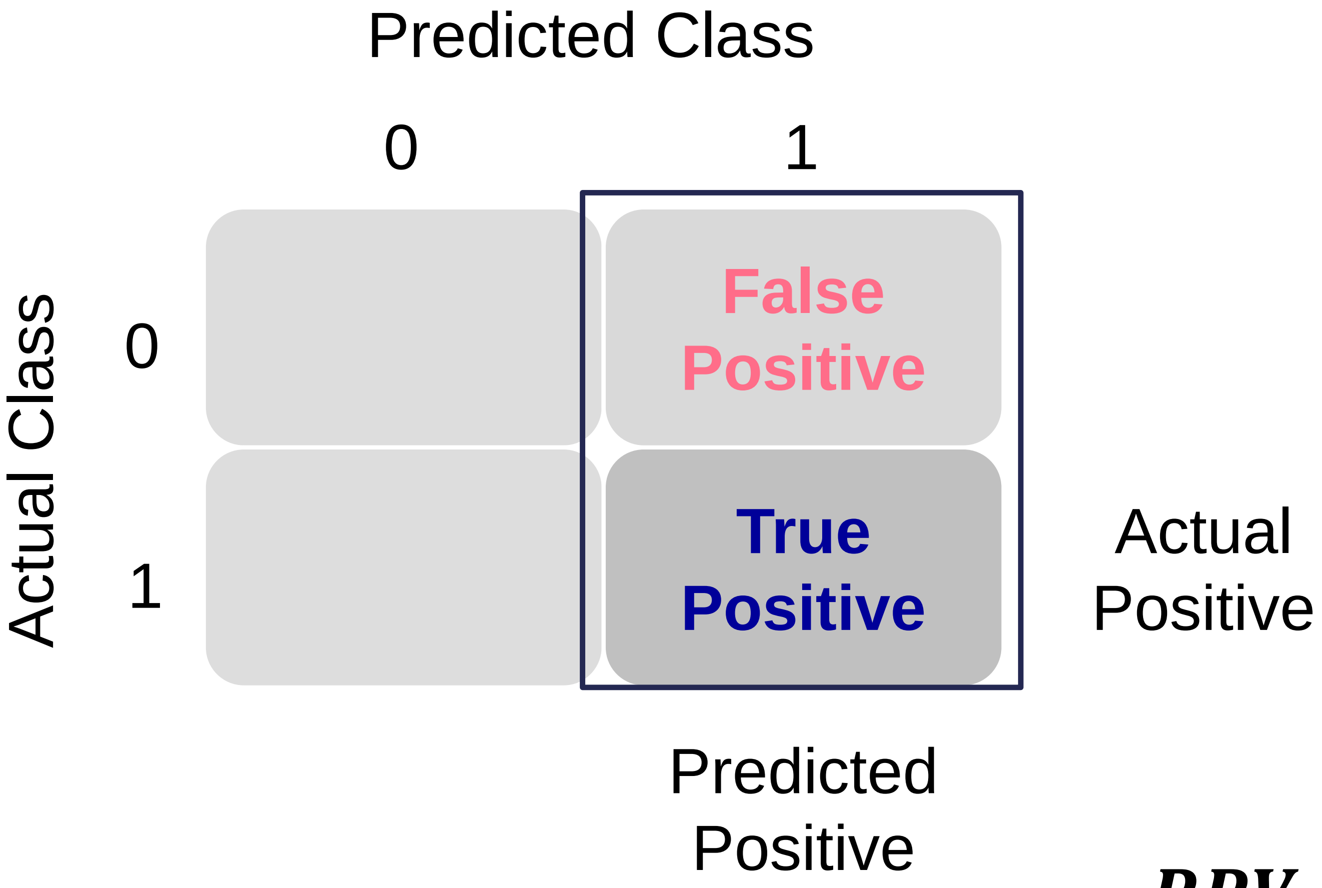
Sensitivity / Recall

Predicted Class

		0	1	
Actual Class	0			
	1	**False Negative**	**True Positive**	Actual Positive
			Predicted Positive	

$$\boldsymbol{TPR = \frac{TP}{TP + FN}}$$

Precision



$$PPV = \frac{TP}{TP + FP}$$

Best Cut-off?

- **Always** consider the cost of false positives and false negatives when doing classification.
- When **NOT** considering costs, many different techniques to “optimal” cut-off.
- $\boldsymbol{F_1}$ **score** (precision-recall version of Youden’s Index):

$$F_1 = 2\left(\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}\right)$$

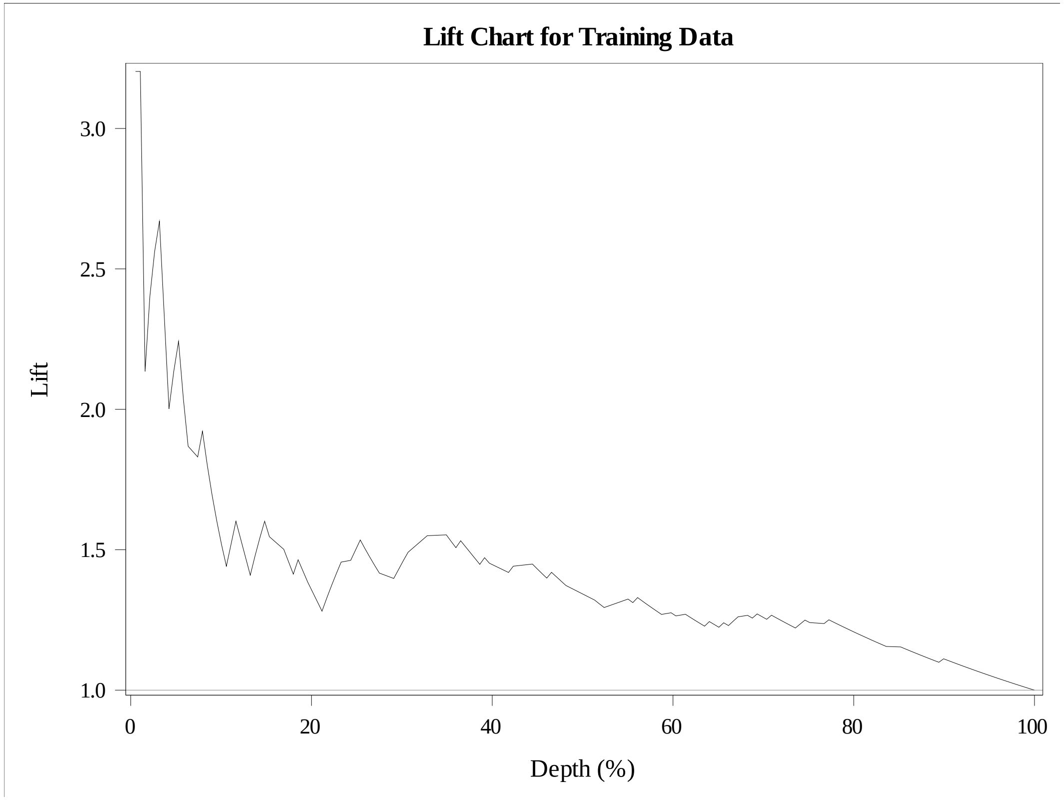
- “Optimal” – precision and recall are weighed equally, so select cut-off that produces highest F_1 score.

Precision & Lift

$$PPV = \frac{TP}{TP + FP}$$

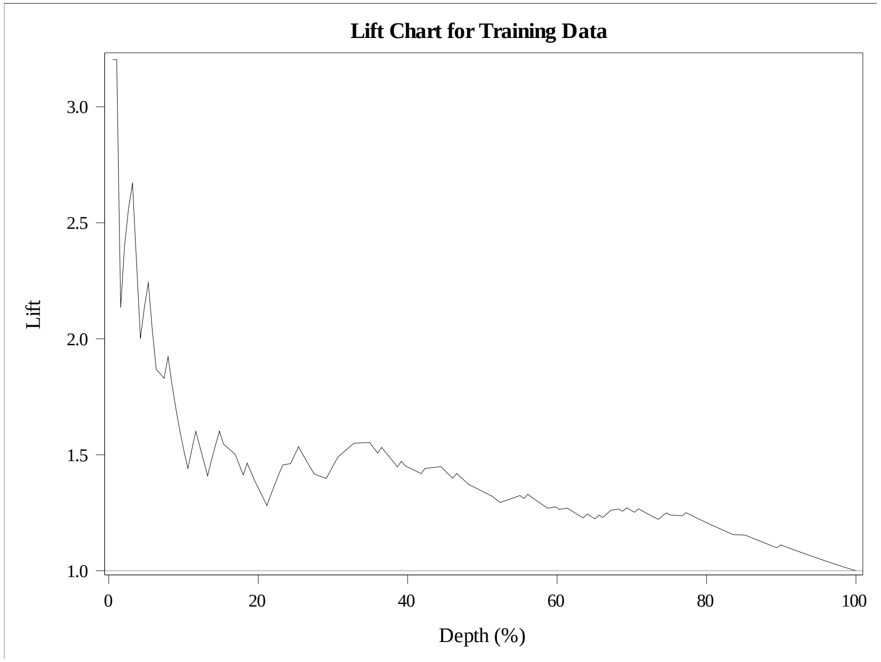
$$PPV = \frac{TP}{Depth}$$

$$Lift = \frac{PPV}{\pi_1}$$



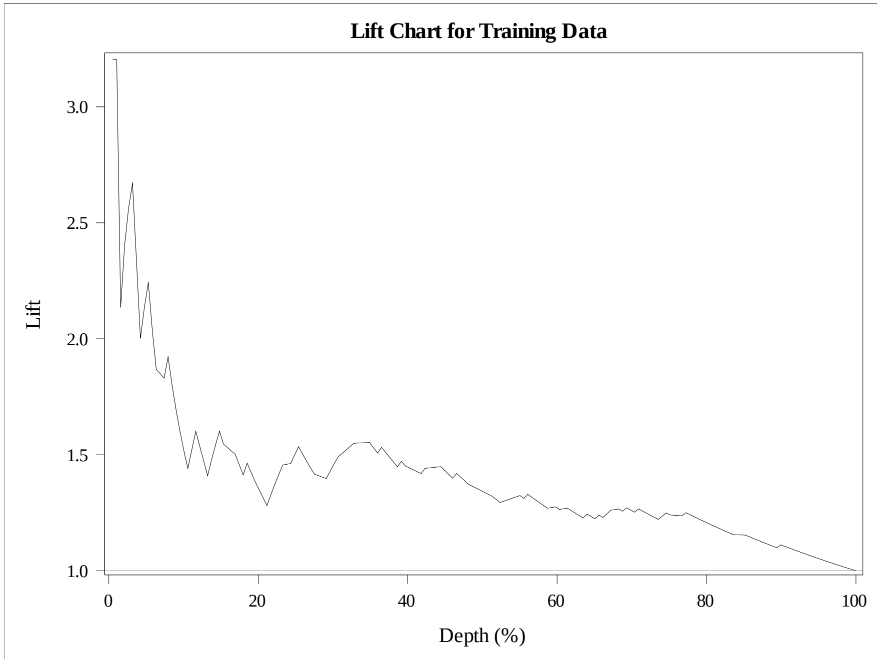
Lift Interpretation

- The top **depth**% of your customers, based on predicted probability, you get **lift** times as many responses compared to targeting a random sample of **depth**% of your customers.



Lift Interpretation

- The top **depth**% of your customers, based on predicted probability, you get **lift** times as many responses compared to targeting a random sample of **depth**% of your customers.
- Careful, in oversampled data, you need to readjust your predicted probabilities!



Precision, Recall, F_1 – SAS

```
data classtable;
   set classtable;
   F1 = 2*(PPV*Sensitivity)/(PPV + Sensitivity);
   drop Specificity NPV Correct;
run;

proc sort data=classtable;
   by descending F1;
run;

proc print data=classtable;
run;
```

Precision, Recall, F_1 – SAS

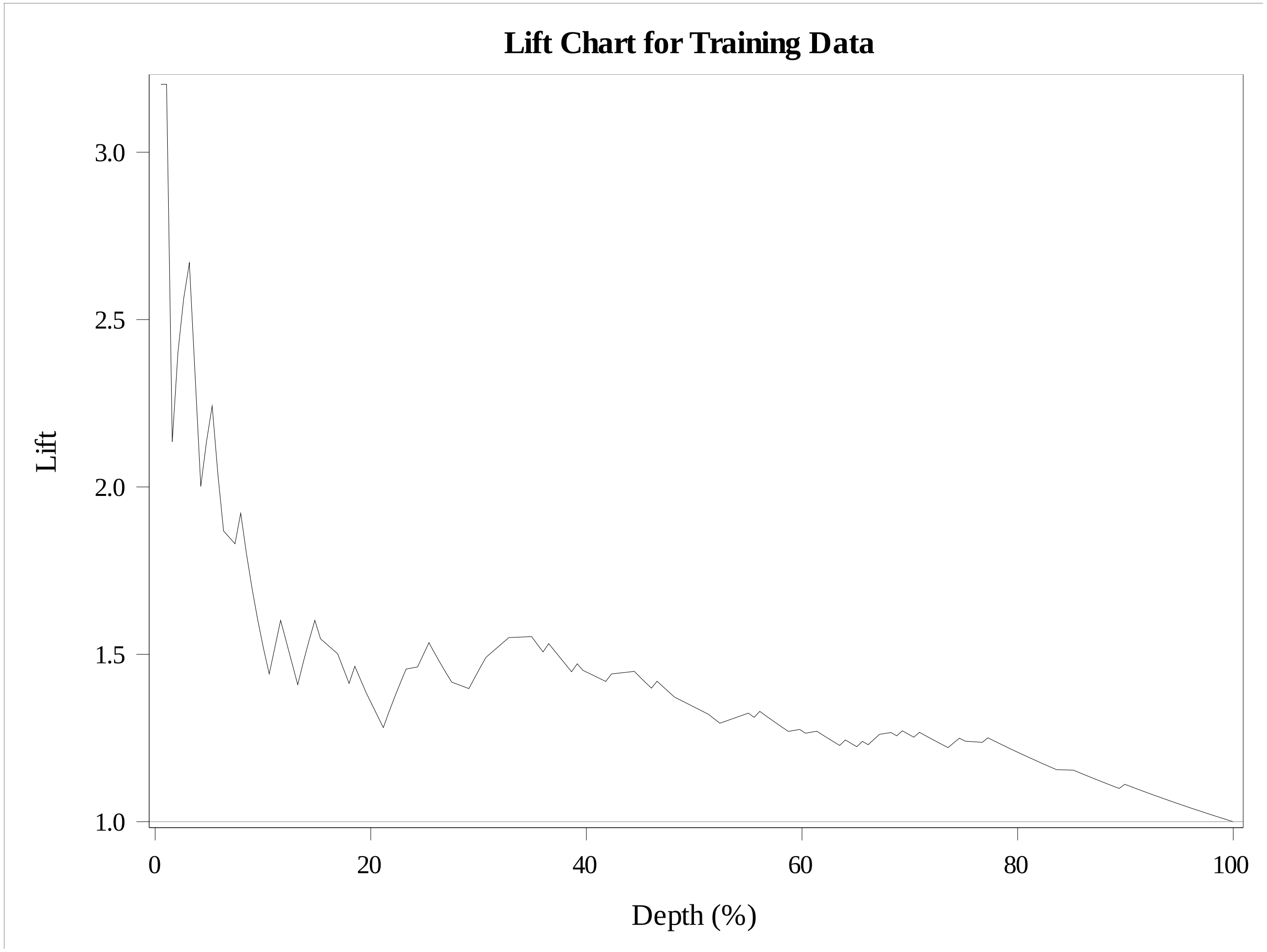
Obs	ProbLevel	Sensitivity	PPV	F1
1	0.200	89.8	38.4	53.8071
2	0.180	89.8	37.9	53.2663
3	0.220	86.4	38.1	52.8497
4	0.160	91.5	36.2	51.9231
5	0.140	96.6	35.4	51.8182
6	0.240	81.4	37.8	51.6129

⋮

Lift Chart – SAS

```
proc logistic data=logistic.lowbwt plots(only)=(oddsratio);
   class race(ref='white') / param=ref;
   model low(event='1') = race lwt smoke / clodds=pl clparm=pl;
   score data=logistic.lowbwt fitstat outroc=roc;
   title 'Modeling Low Birth Weight';
run;
quit;

data work.roc;
   set work.roc;
   cutoff = _PROB_;
   specif = 1-_1MSPEC_;
   depth=(_POS_+_FALPOS_)/189*100;
   precision=_POS_/(_POS_+_FALPOS_);
   acc=_POS_+_NEG_;
   lift=precision/0.3122;
run;
```

Lift Chart – SAS



Precision, Recall, F_1 – R

```
prec <- NULL
reca <- NULL
f1 <- NULL
cutoff <- NULL

for(i in 1:49){
  cutoff = c(cutoff, i/50)
  reca <- c(reca, sensitivity(bwt$low, bwt$p_hat,
              threshold = i/50))
  prec <- c(prec, precision(bwt$low, bwt$p_hat,
              threshold = i/50))
  f1 <- c(f1, 2*((prec[i]*reca[i])/(prec[i]+reca[i])))
}

ctable <- data.frame(cutoff, reca, prec, f1)
```

Precision, Recall, F_1 – R

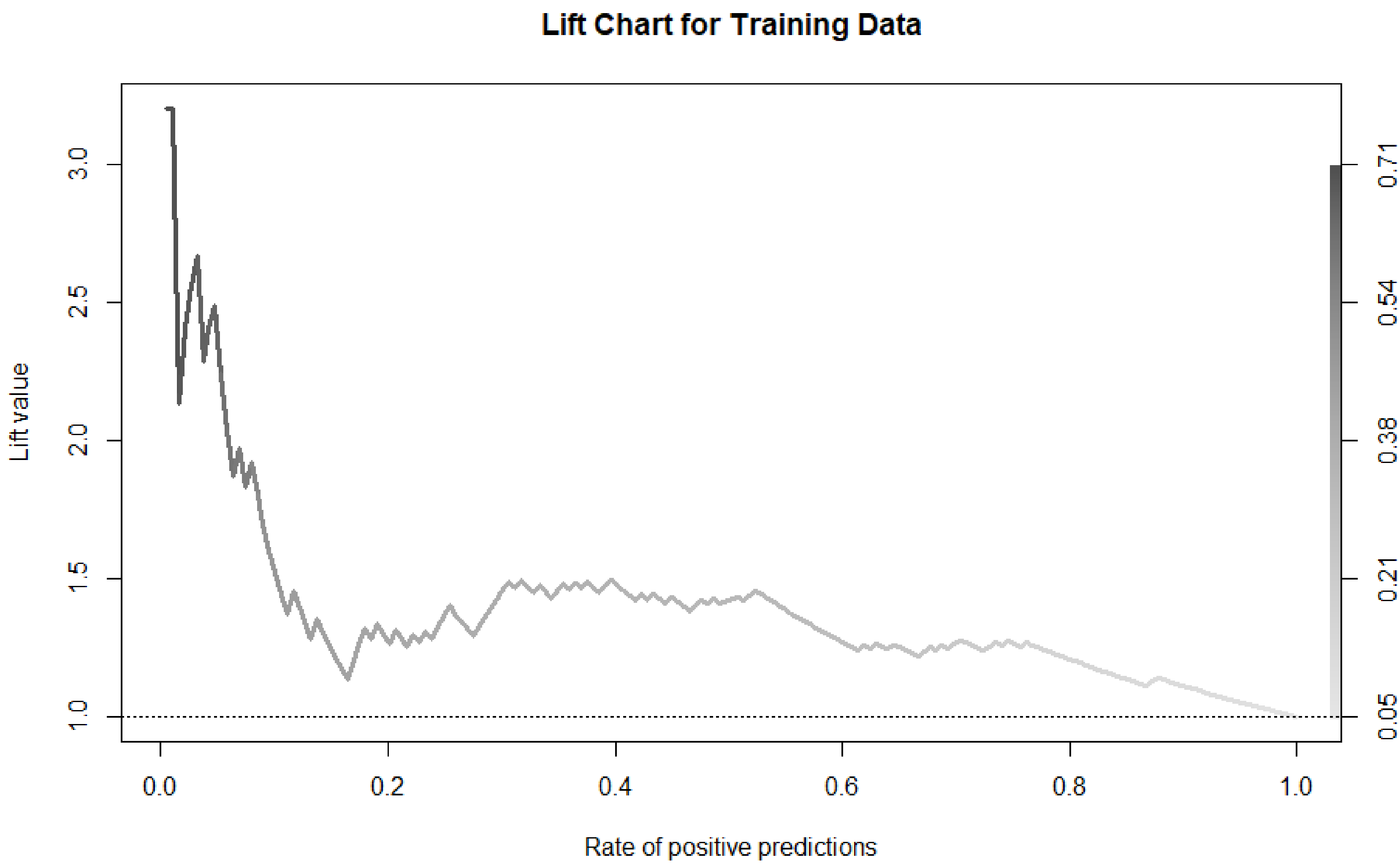
```
##    cutoff       reca      prec         f1
## 1    0.02 1.00000000 0.3121693 0.47580645
## 2    0.04 1.00000000 0.3121693 0.47580645
## 3    0.06 1.00000000 0.3138298 0.47773279
## 4    0.08 1.00000000 0.3189189 0.48360656
## 5    0.10 1.00000000 0.3314607 0.49789030
## 6    0.12 1.00000000 0.3410405 0.50862069
## 7    0.14 0.96610169 0.3630573 0.52777778
## 8    0.16 0.96610169 0.3800000 0.54545455
## 9    0.18 0.93220339 0.3873239 0.54726368
## 10   0.20 0.89830508 0.3868613 0.54081633
## 11   0.22 0.89830508 0.3925926 0.54639175
## 12   0.24 0.86440678 0.3953488 0.54255319
## 13   0.26 0.81355932 0.3870968 0.52459016
```

⋮

Lift Chart – R

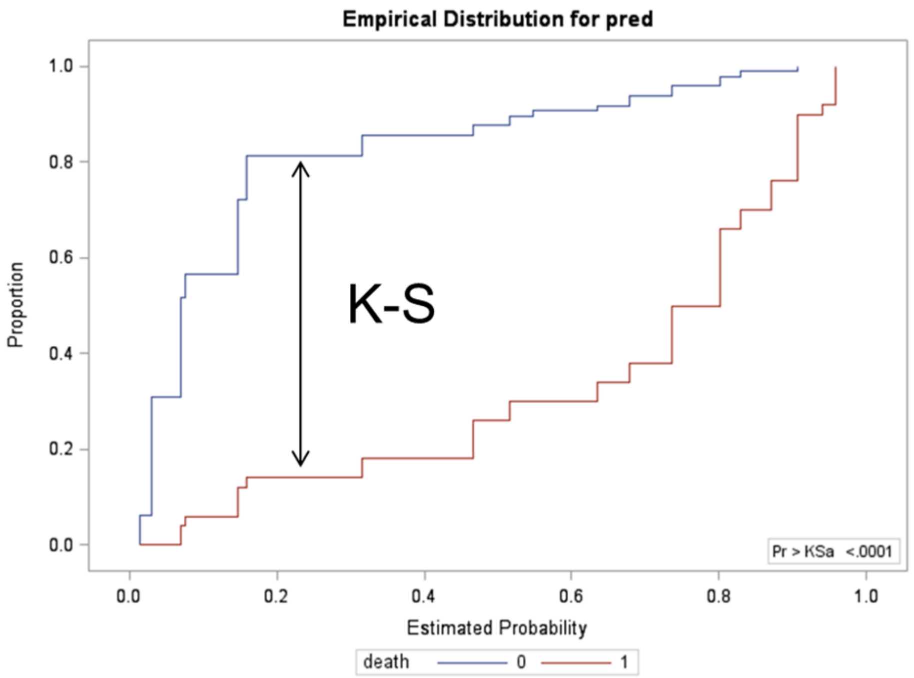
```
perf <- performance(pred, measure = "lift", x.measure = "rpp")
plot(perf, lwd = 3, colorize = TRUE, colorkey = TRUE,
     colorize.palette = rev(gray.colors(256)),
     main = "Lift Chart for Training Data")
abline(h = 1, lty = 3)
```

Lift Chart – R


Lift Chart for Training Data
Lift value
3.0
2.5
2.0
1.5
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Rate of positive predictions
0.71
0.54
0.38
0.21
0.05

K-S Statistic

- The Two-Sample K-S statistic can determine if there is a difference between two cumulative distribution functions.
- Has a corresponding hypothesis test, with D test statistic, and p-value.



Best Cut-off?

- **Always** consider the cost of false positives and false negatives when doing classification.
- When **NOT** considering costs, many different techniques to "optimal" cut-off.
- $\boldsymbol{KS}$ **statistic D** (maximum difference between TPR and FPR):

$$D = \max_{depth}(TPR - FPR)$$

- "Optimal" – select cut-off that produces highest D statistic.

K-S Statistic – SAS

```
proc logistic data=logistic.lowbwt noprint;
   class race(ref='white') / param=ref;
   model low(event='1') = race lwt smoke;
   output out=predprobs p=phat;
run;

proc npar1way data=predprobs d plot=edfplot;
   class low;
   var phat;
run;
```

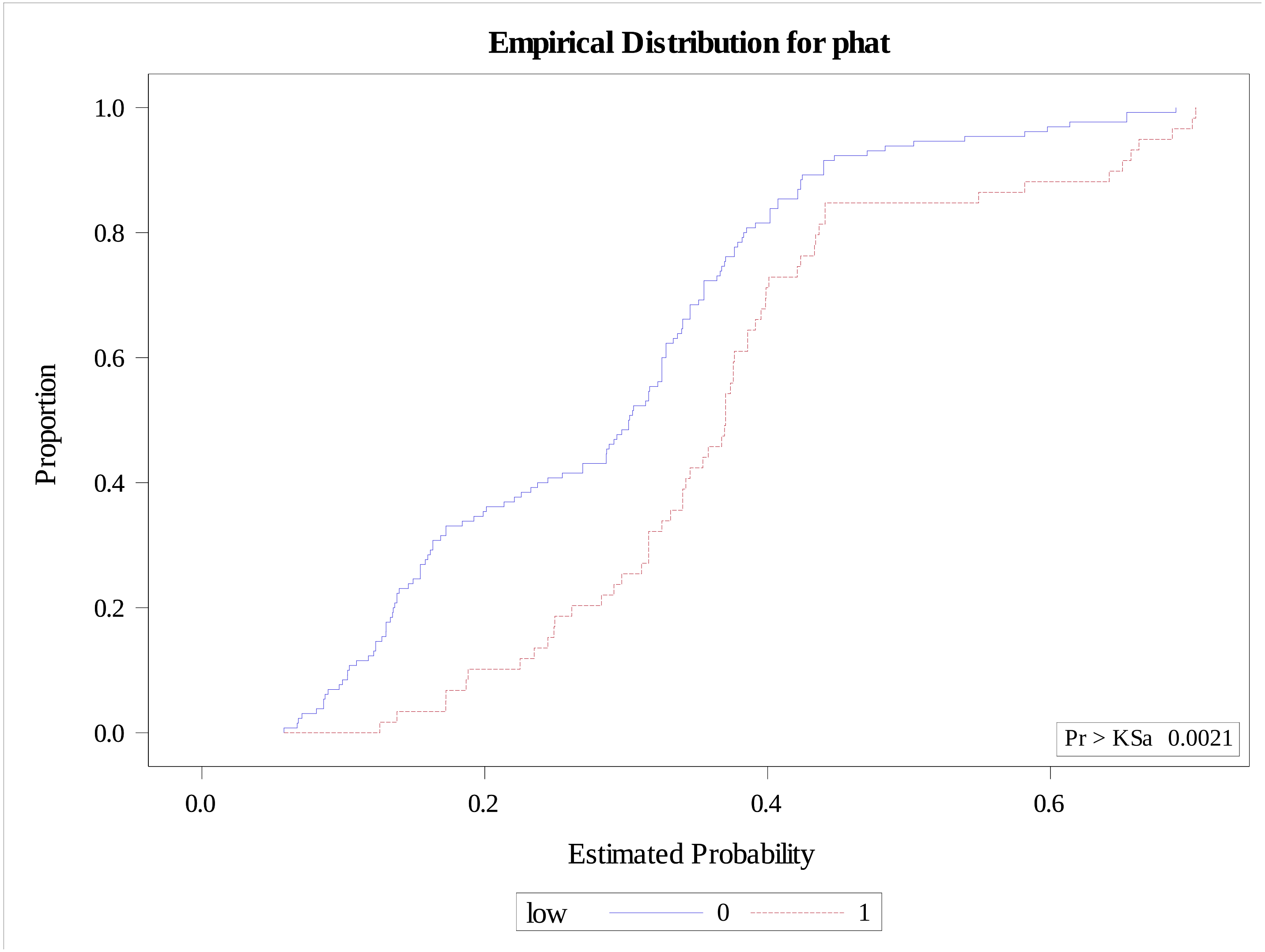
K-S Statistic – SAS

The NPAR1WAY Procedure

Kolmogorov-Smirnov Test for Variable phat Classified by Variable low			
low	**N**	**EDF at Maximum**	**Deviation from Mean at Maximum**
0	130	0.646154	1.032979
1	59	0.355932	-1.533336
Total	189	0.555556	
Maximum Deviation Occurred at Observation 27			
Value of phat at Maximum = 0.339202			

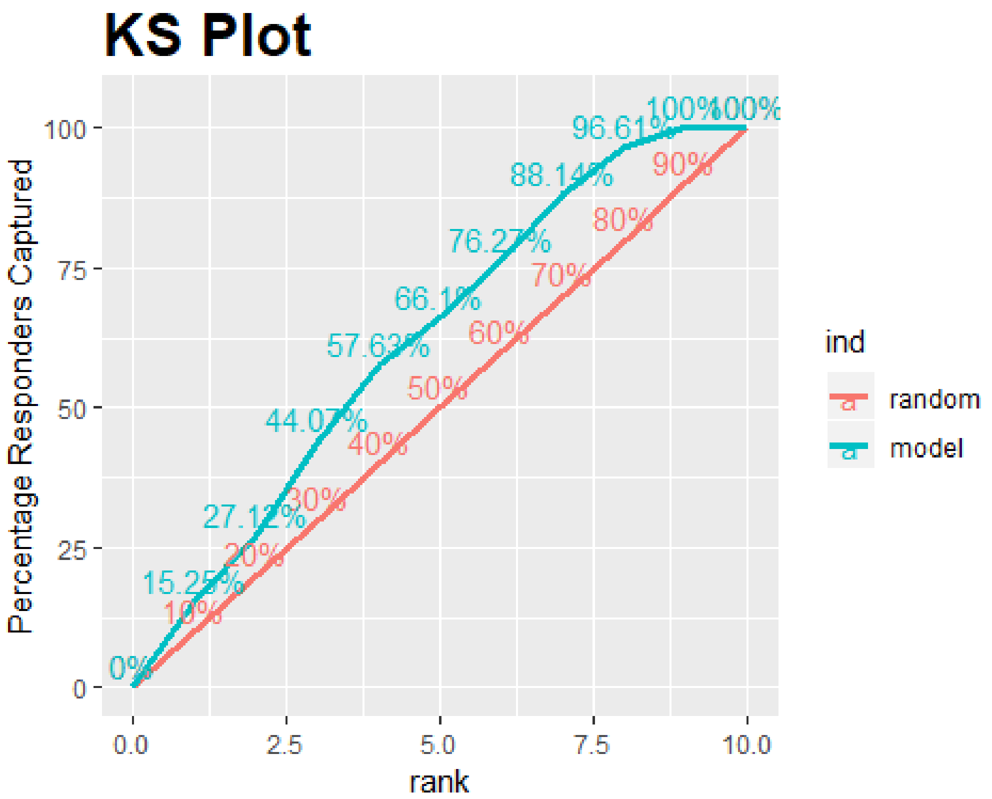
Kolmogorov-Smirnov Two-Sample Test (Asymptotic)	
D = max |F1 - F2|	0.2902
Pr > D	0.0021

K-S Statistic – SAS



K-S Statistic – R

```
ks_stat(bwt$low, bwt$p_hat)
## [1] 0.2583
ks_plot(bwt$low, bwt$p_hat)
```



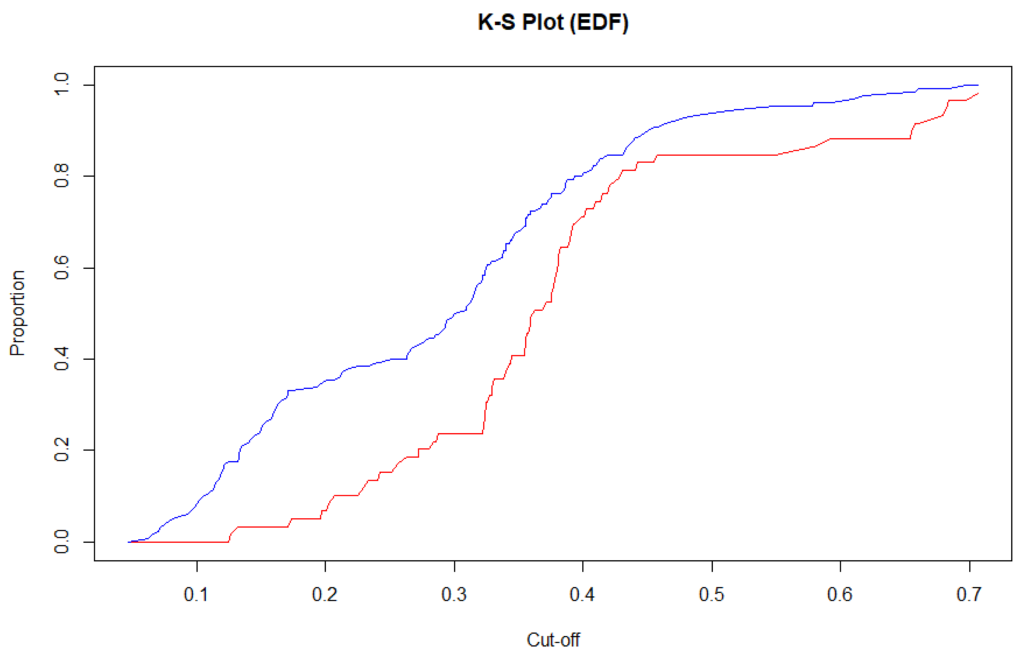
K-S Statistic – R

```
perf <- performance(pred, measure = "tpr", x.measure = "fpr")
KS <- max(perf@y.values[[1]] - perf@x.values[[1]])
cutoffAtKS <- unlist(perf@alpha.values)[which.max(perf@y.values[[1]]
                     - perf@x.values[[1]])]
print(c(KS, cutoffAtKS))
```

```
## [1] 0.2902216 0.3399442
```

K-S Statistic – R

```
plot(x = unlist(perf@alpha.values), y = (1-unlist(perf@y.values)),
     type = "l", main = "K-S Plot (EDF)",
     xlab = 'Cut-off',
     ylab = "Proportion",
     col = "red")
lines(x = unlist(perf@alpha.values), y = (1-unlist(perf@x.values)),
      col = "blue")
```



Accuracy

		Predicted Class		
		0	1	
Actual Class	0	**True Negative**	**False Positive**	Actual Negative
	1	**False Negative**	**True Positive**	Actual Positive
		Predicted Negative	Predicted Positive	

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

Accuracy

Predicted Class

Actual Class	0	1	
0	**True Negative**	**False Positive**	Actual Negative
1	**False Negative**	**True Positive**	Actual Positive
	Predicted Negative	Predicted Positive	

$$\boldsymbol{Accuracy = \frac{TP + TN}{n}}$$

Misclassification (Error) Rate

		Predicted Class		
		0	1	
Actual Class	0	True Negative	False Positive	Actual Negative
	1	False Negative	True Positive	Actual Positive
		Predicted Negative	Predicted Positive	

$$Error = \frac{FP + FN}{n}$$

Accuracy and Error

- Accuracy and error can be easily fooled so careful focusing only on them.
- If your data has 10% events and 90% non-events, you can have a 90% accurate model by guessing non-events for **every** observation.
- There is more to model building than simply maximizing overall classification accuracy.
- Good numbers to report, but not necessarily to choose models on.

Closing Thoughts on Classification

- Classification is a **decision** that is extraneous to statistical modeling.
- Although logistic regression tends to work well in classification, it is a **probability model** and does not output 1's and 0's.
- Classification assumes cost for each individual is the same.
  - Useful for groups.
  - Careful about single observation decisions.