Logistic Modeling without Split-Samples

BRUCE LUND
Statistical Trainer, Novi, MI

Blundz@iCloud.com … or
Blund.data@gmail.com

v1

Double Dipping

N. Kriegeskorte, et. al. (2009) wrote this:

"Double Dipping is the use of the same dataset for selection and selective analysis. It gives distorted descriptive statistics and invalid statistical inference"

"Circular analysis in systems neuroscience: the dangers of double dipping", (2009) *Nature Neuroscience* ...

Double Dipping could arise if fitting a Logistic Model to TRAINING and using the TRAINING dataset over again for computing model validation statistics (e.g. c-statistic, average squared error, etc.).

Can Double Dipping be avoided without a split-sample?
... That is, without a separate sample for VALIDATION

It is a purpose of the Talk to answer this question.

Dr. Daniela Witten during a 2022 webinar hosted by Wake Forest University conjectured that the Kriegeskorte paper (cited above) was the first usage of "double dipping" as a statistical term

Don't Double Dip … Data or Chips

Dr. Witten was incorrect.
We must remember the Seinfeld episode of 1993 where George double-dipped a chip!

Picture Removed

A little bit of humor
to start off the talk !

Optimism Correction to replace Split-Sampling?

Split-Sampling

Can the problem of double dipping be avoided without a split-sample?

- If so, this would have the advantage of fitting the logistic model on the entire Analysis dataset, giving better predictor variable selection and better coefficient estimation.
- But this leaves open the question of how to perform model validation.
- It is a purpose of the paper to show how split-sampling can be avoided.

Optimism Correction with Bootstrap Sampling

The Model is fitted on the Analysis dataset (i.e. all the data)

- The performance metrics are also computed on the Analysis dataset (double dipping!!)
- But these performance metrics are then corrected by an "optimism correction".
- This talk explains how *bootstrap sampling* is utilized in finding the optimism correction.

➢ Another alternative to split-sampling is Cross-Validation. This is not discussed today.

Bootstrap Sampling … how to do it with SURVEYSELECT

Suppose dataset BOOT has 5 observations.
e.g. if BOOT = {0, 1, 2, 3, 4}, then one possible bootstrap sample is {0, 0, 1, 3, 4}

A bootstrap sample from BOOT is formed by 5 random picks from BOOT with Replacement.

PROC SURVEYSELECT can perform bootstrap sampling. Here are 2 bootstrap samples from BOOT:

```
DATA BOOT;
   DO X = 0 to 4;
   OUTPUT;
   END;
PROC SURVEYSELECT DATA=BOOT
OUT=BootSamples
NOPRINT /* Don't print a summary of sampling */
SEED=111
METHOD=URS /* with replacement */
SAMPRATE=1 /* Sample size = 100% (size of Boot) */
REPS=2 /* Create two bootstrap samples */
;
PROC PRINT DATA=BootSamples;
run;
```

Obs	Replicate	X	NumberHits
1	1	0	1
2	1	1	1
3	1	2	1
4	1	3	1
5	1	4	1
6	2	0	1
7	2	2	1
8	2	3	3

← Sheer Luck !!
Seed = 111

NEXT

Logistic Model Performance Metrics

Widely used performance metrics to measure a Logistic Model include:

- The c-statistic (also called area under the ROC curve … AUROC)
- Average Square Error
- Misclassification Rate

The 3 metrics mentioned above are "global metrics" … referring to the overall model fit

Another metric, which is widely used in *target marketing,* is the Lift Chart

- Lift Chart measures ability of the model's P's (model probabilities) to:
  - Estimate average event rates for small intervals of P's … more discussion to follow.

This talk will focus on the Lift Chart (and as well, the c-statistic)

So, what is a Lift Chart and how does Optimism Correction apply to it ??

… See next slides.

Here is WORK, the dataset for the example of this Talk

1000 observations with 50% events.
Two predictors X1, X2

```
%LET SEED = 3; %LET N = 1000;
DATA WORK;
call streaminit(&SEED);
DO ID = 1 to &N;
 if mod(ID,2) = 0 then Y = 1;
 else Y = 0;
 if Y = 1 then do;
    X1 = rand('normal') + .2;
    X2 = rand('normal') + .2;
    end;
 if Y = 0 then do;
    X1 = rand('normal') - .2;
    X2 = rand('normal') - .2;
    end;
 output;
 end;
run;
```

Use SURVEYSELECT to split WORK into

- Train and Validate
- With 500 observations in each dataset
- Each with event rate = 50%

See the Next Slide

Here is WORK, the dataset for the example of this Talk

```
PROC SORT DATA = WORK; BY Y;
PROC SURVEYSELECT DATA = WORK
OUT = WORK(rename=(SELECTED=PART))
OUTALL
METHOD = SRS /* no replacement */
N = (250 250) /* leaves 250 Y=1, 250 Y=0 */
SEED = 1;
STRATA Y;
run;
```

Before SURVEYSELECT, Sort by Y (STRATA variable)

OUTALL causes ALL observations to be Output

N=(250 250) codes 250 non-events, 250 events as SELECTED=1

I did a rename: SELECTED=**PART**.
I used **PART**=1 for the Validation dataset.

What remains is "unselected"
250 events and 250 non-events where SELECTED=0
After rename as **PART**=0 … these are the Train dataset

Use SURVEYSELECT … Not this "short-cut" for Train/Validate split

```
%LET SEED = 6;
DATA WORK; SET WORK;
call streaminit(&SEED);
if rand('uniform') < 0.5 then PART = 0;
      else PART = 1;
```

Observations are assigned to Train or Validate based on random "coin flips"

The event-rate for Train will not equal the event-rate for Validate.

Dataset	Freq	P01	P99	P_mean	Y_mean
Train	488	0.271	0.712	0.488	0.488
Validate	512	0.271	0.733	0.496	0.512

← This is what happened.

This causes problems when validating the model.

- Model is fitted on TRAIN with event-rate = 0.488
- Validation dataset is scored using the Model from Train
- Scored P's on Validation will undershoot the Validation event-rate = 0.512.

NEXT

What is a Lift Chart?

At ALL P_mean = Y_mean by MLE solution

- Fit a Logistic Model to Train (i.e. PART = 0)
- P (Model's probability) ranks the Obs. from the Logistic Model into Groups. Here, 10 groups ➔
- Highest P's are in Rank_P = 0. Lowest in Rank_P = 9.
- In this example, each Rank has 50 Obs.
- P_mean is the mean of P in the rank
- Y_mean is the event-rate (mean of Y) in rank
- Lift: For Rank_P=0, the LIFT=0.68 / 0.5 = 136%

Lift Chart for Train

Rank_P	Freq	P_mean	Y_mean	_Lift_
ALL	500	0.5	**0.5**	
0	50	0.675	**0.680**	**136%**
1	50	0.609	0.540	108%
2	50	0.568	0.580	116%
3	50	0.539	0.640	128%
4	50	0.513	0.500	100%
5	50	0.489	0.540	108%
6	50	0.464	0.480	96%
7	50	0.432	0.320	64%
8	50	0.388	0.360	72%
9	50	0.323	0.360	72%

See paper for LIFT CHART SAS coding

These 10 Ranks (deciles) are *inconsistent* because:

- Column _Lift_ is not Monotonic (poor discrimination).
- Poor Calibration (disagreement) of Y_mean and P_mean … see Rank_P = 3 and 7

➔ A precise meaning for "Calibration" is given on the next slide.

Here is a way to quantitatively measure calibration

Consider P_Rank = 3

A pseudo 80% confidence interval for "true" Event Rate for this rank is: …

0.64 +/- 1.28 * SD = (0.553, 0.727)

where SD = sqrt(Y_mean * (1-Y_mean) / Freq) = sqrt (0.64 * (1-0.64) / 50) = 0.068

Lift Chart for Train

Rank_P	Freq	P_mean	Y_mean	Low	High	P_In_Out
ALL	500	0.5	0.5			
0	50	0.675	0.68	0.596	0.764	IN
1	50	0.609	0.54	0.450	0.630	IN
2	50	0.568	0.58	0.491	0.669	IN
3	50	0.539	0.64	0.553	0.727	OUT
4	50	0.513	0.50	0.409	0.591	IN
5	50	0.489	0.54	0.450	0.630	IN
6	50	0.464	0.48	0.390	0.570	IN
7	50	0.432	0.32	0.236	0.404	OUT
8	50	0.388	0.36	0.273	0.447	IN
9	50	0.323	0.36	0.273	0.447	IN

P_mean estimates the true Event Rate.

Does P_mean = 0.539 fall inside (0.553, 0.727) …? … OUT

So, P_In_Out = "OUT" for P_Rank= 3

This is poor calibration

Likewise for Rank_P = 7 … "OUT"

Otherwise: "IN"

How to Fix this Lift Chart ?

Ad hoc combining of levels ?? … maybe this:

Combine ranks 1, 2, 3.
Combine ranks 4, 5.
Combine ranks 6, 7.

- This is messy and rather arbitrary
- New ranks have unequal frequencies
- Hides the underlying issue which is:

  Model does not have sufficient calibration and discrimination to support 10 ranks

Lift Chart for Train

Rank_P	Freq	P_mean	Y_mean	_Lift_
ALL	500	0.5	0.5	
0	50	0.675	0.680	136%
1	50	0.609	0.540	108%
2	50	0.568	0.580	116%
3	50	0.539	0.640	128%
4	50	0.513	0.500	100%
5	50	0.489	0.540	108%
6	50	0.464	0.480	96%
7	50	0.432	0.320	64%
8	50	0.388	0.360	72%
9	50	0.323	0.360	72%

Better approach: Find a number of Ranks that will give a *Consistent* Lift Chart

I have a "seat-of-the-pants" formula in my paper. I'll skip the details here.

This formula says to use 4 ranks.

NEXT

My formula says to use 4 ranks

Lift Chart for Train with 4 Ranks

Rank_P	Freq	P01	P99	P_mean	Y_mean	_Diff_	_Lift_	Low	High	P_In_Out
ALL	500	0.270	0.733	0.500	0.500	-.000				
0	125	0.566	0.754	0.629	0.616	-.013	123%	0.545	0.687	IN
1	125	0.502	0.566	0.532	0.560	0.028	112%	0.487	0.633	IN
2	125	0.435	0.501	0.470	0.480	0.010	96%	0.407	0.553	IN
3	125	0.224	0.433	0.369	0.344	-.025	69%	0.274	0.414	IN

Consistent: Calibration: all P_mean are "IN" and _Lift_ is monotonic.

Top _Lift_ is 123% … is this good? Depends on context:

- ✓ Are sales to prospects in top rank "incremental" (i.e. occur because of marketing campaign)
- ✓ What is the profit from an incremental sale?

But not discussing incrementality today (… requires a control group and a treated group)

- ✓ Lift depends on the choice of Population.

  (If "wide" population with weak prospect (unlikely to buy), then Y_mean for "ALL" is much lower and Lift for Rank_P = 0 becomes higher.)

How Many Ranks for the Validation Lift Chart?

Continue with 4 ranks, as for Lift Chart for Train

Lift Chart for Validation with 4 Ranks

Rank_P	Freq	P01	P99	P_mean	Y_mean	_Diff_	_Lift_	Low	High	P_In_Out
ALL	500	0.267	0.736	0.497	0.500	0.003				
0	125	0.563	0.767	0.626	0.624	-.002	125%	0.553	0.695	IN
1	125	0.497	0.561	0.527	0.552	0.025	110%	0.479	0.625	IN
2	125	0.433	0.495	0.464	0.504	0.040	101%	0.431	0.577	IN
3	125	0.234	0.431	0.373	0.320	-.053	64%	0.252	0.388	IN

Commentary:

- P_mean, at 0.497, is very near to the Y_mean of 0.5, so there is overall consistency.
- All values of P_In_Out are "IN".
- The _Lift_ column is monotonic.
- Top _Lift_ value (125%) is very similar to top _Lift_ from Lift Chart on Train (123%)

So, is everything Good ?

NO !! … see next slides

Generate 20 split-samples

Could a particular random assignment of Train and Validation, by chance, give substantially different results when computing the Lift Chart on Validation?

Here is an experiment: Use SEED's in SURVEYSELECT from 1 to 20

```
PROC SORT DATA = WORK; BY Y;
PROC SURVEYSELECT DATA = WORK
OUT = WORK(rename=(SELECTED=PART)) OUTALL
METHOD = SRS /* sampling without replacement */
N = (250 250)
SEED = 1;  /* ← rerun with 2, …, 20 */
STRATA Y;
run;
```

The logistic model from SEED = 1 is the model that has been discussed so far.

The other 19 splits will show how the vagaries of random chance can sometimes lead to unsatisfactory modeling results.

Results for Seeds 1 - 20

← Started Here: Seed = 1

	Train			Validation		
Seed	Calib	Mono	Lift: Hi / Low	Calib	Mono	Lift: Hi / Low
1	**Y**	**Y**	**1.23 / 0.69**	**Y**	**Y**	**1.25 / 0.64**
2	Y	Y	1.30 / 0.66	N	Y	1.18 / 0.67
3	Y	Y	1.18 / 0.66	Y	Y	1.25 / 0.66
4	N	N	1.14 / 0.70	N	Y	1.28 / 0.66
5	Y	Y	1.22 / 0.64	Y	Y	1.22 / 0.67
6	Y	Y	1.33 / 0.74	N	N	1.10 / 0.59
7	Y	Y	1.18 / 0.69	Y	Y	1.25 / 0.67
8	Y	Y	1.17 / 0.66	Y	Y	1.30 / 0.66
9	N	N	1.15 / 0.69	N	N	1.26 / 0.64
10	Y	Y	1.28 / 0.58	Y	Y	1.23 / 0.77
11	Y	Y	1.22 / 0.74	N	Y	1.26 / 0.64
12	Y	Y	1.26 / 0.64	Y	Y	1.23 / 0.70
13	Y	N	1.09 / 0.70	N	Y	1.34 / 0.62
14	Y	Y	1.28 / 0.66	N	N	1.14 / 0.66
15	N	N	1.20 / 0.66	Y	Y	1.20 / 0.69
16	Y	Y	1.38 / 0.69	N	N	1.14 / 0.69
17	Y	Y	1.36 / 0.59	N	N	1.07 / 0.77
18	Y	Y	1.30 / 0.66	N	Y	1.17 / 0.67
19	Y	Y	1.26 / 0.62	N	Y	1.18 / 0.70
20	Y	Y	1.31 / 0.61	N	Y	1.17 / 0.74

"N" is an unsatisfactory outcome

For seeds 4, 9, 13, 15 the modeler might abandon the effort.

Modeler might conclude that a promising model failed to validate for: 6, 14, 16, 17

It turns out that Model1 is something of an anomaly.

Model1 was just good luck.

The Alternative to Split-Sampling: Optimism Correction

- Suppose N ~ 20,000, event-rate ~ 0.5, and ~ 50 predictors?

  It is probable that Lift Chart Validation results would not depend, noticeably, on a particular split.

- BUT: Prior Slide is a cautionary tale for usage of split-sampling when Analysis Dataset is small to medium size such as N ~ 1000 with event-rate ~ 0.5 (even with only 2 X's).

An alternative to split-sampling is "Optimism Correction"

See next slides …

NEXT

For Optimism Correction: First Step is to fit Apparent Model

FIRST, fit the Model to entire Analysis Dataset.

This is called: Apparent Model

For dataset WORK with N = 1000, the Apparent Model is simply:

```
PROC LOGISTIC DATA = WORK desc;
MODEL Y = X1 X2;
SCORE DATA = WORK;
```

Performance Metric will be a **Lift Chart** … but other metrics (c-stat, etc.) can be used.

Rank_P	Freq	P01	P99	P_mean	Y_mean	_Diff_	_Lift_	Low	High	P_In_Out
ALL	1000	0.252	0.751	0.500	0.500	-.000				
0	250	0.573	0.773	0.637	0.604	-.033	121%	0.564	0.644	IN
1	250	0.500	0.572	0.534	0.568	0.034	114%	0.528	0.608	IN
2	250	0.430	0.499	0.465	0.496	0.031	99%	0.456	0.536	IN
3	250	0.209	0.428	0.363	0.332	-.031	66%	0.294	0.370	IN

Performance of the Apparent Model with Optimism Correction

Step 1: From WORK: 200 (typically) bootstrap samples are created. Here is the code.

```
PROC SURVEYSELECT DATA=WORK OUT=BootSamples
NOPRINT SEED=111 METHOD=URS /* with replacement */
SAMPRATE=1 /* Sample = 100% */ REPS=200; /* Create 200 Samples */
```

- For each of the 200 samples, a logistic model is fit with Y and X1, X2.
- 200 Models and Lift Charts are saved. The 200 Lift Charts are called LC_{boot}

Step 2: For each bootstrap model (Step 1), WORK is scored by the model and Lift Charts are computed (200). These will be called LC_{full}.

So far: 200 Lift Charts from the bootstrap samples … LC_{boot}
200 Lift Charts from scoring WORK with the bootstrap models … LC_{full}

Step 3a: Now take average of 200 Lift Charts from the bootstrap samples

Step 3b: Now take average of 200 Lift Charts from scoring WORK with bootstrap models

Measure Performance of the Apparent Model … more

Step 4: Optimism for Lift Chart is $LC_{optimism}$ = average(LC_{boot}) - average(LC_{full})

i.e., difference of averages.

Step 5: Now, the Optimism Corrected Lift Chart is

$$LC_{corrected} = LC_{apparent} - LC_{optimism}$$

This is the Lift Chart that should be reported as the Lift Chart performance of the Apparent model.

- - - - - - -

Now, do the calculations of the Optimism Corrected Lift Chart for WORK

See next slides.

Measure Performance of Apparent Model on WORK

The next slides will apply Optimism Correction process to the model fitted to WORK.

```
PROC LOGISTIC DATA = WORK desc;
MODEL Y = X1 X2;
SCORE DATA = WORK;
```

Here is the Lift Chart for Apparent Model:

Rank_P	Freq	P01	P99	P_mean	Y_mean	_Diff_	_Lift_	Low	High	P_In_Out
ALL	1000	0.252	0.751	0.500	0.500	-.000				
0	250	0.573	0.773	0.637	0.604	-.033	121%	0.564	0.644	IN
1	250	0.500	0.572	0.534	0.568	0.034	114%	0.528	0.608	IN
2	250	0.430	0.499	0.465	0.496	0.031	99%	0.456	0.536	IN
3	250	0.209	0.428	0.363	0.332	-.031	66%	0.294	0.370	IN

200 Bootstrap samples and Models (averaged results)

Bootstrap Model		
RANKP	P_mean	Y_mean
ALL	0.49966	0.49966
0	0.63901	0.61595
1	0.53505	0.56305
2	0.46421	0.48807
3	0.36058	0.33179

−

Scored Bootstrap on Full Data		
RANKP	P_mean	Y_mean
ALL	0.49960	0.5
0	0.63947	0.61238
1	0.53475	0.56174
2	0.46382	0.48652
3	0.36036	0.33936

=

Optimism = Boot - Full		
RANKP	P_mean	Y_mean
ALL	0.00006	-0.00034
0	-0.00046	0.00357
1	0.0003	0.00131
2	0.00039	0.00155
3	0.00022	-0.00757

- Optimism Corrected gives slightly less Lift than does the Apparent Model
- Chance variation has been averaged out … in contrast with a single split-sample result
- This is the Lift Chart that should be reported

Apparent				
Rank_P	P_mean	Y_mean	Lift	IN/OUT
ALL	0.500	0.500		
0	0.637	0.604	121%	IN
1	0.534	0.568	114%	IN
2	0.465	0.496	99%	IN
3	0.363	0.332	66%	IN

Optimism Corrected = Apparent - Optimism				
RANKP	P_mean	Y_mean	Lift	IN/OUT
ALL	0.500	0.500		
0	0.637	0.600	120%	IN
1	0.534	0.567	113%	IN
2	0.465	0.494	99%	IN
3	0.363	0.340	68%	IN

Disclosure: Difficulties when using Optimism Correction

Modeler might have a predictor X and may wonder whether to include X^2 in the model.

… Might use data analysis to decide whether to include X^2 in the model (looking X v. Y)

… Then, the model is fitted (using the decision made above)

This is Double Dipping and would invalidate optimism correction …

… unless this "data analysis" can be built into optimism correction process..

Here is a way to handle the "X^2 decision" and not Double Dip

… (also suppose there are X1-X5 and class C1-C2 as candidate variables)

First, fit Apparent Model with the `HIERARCHY=single` option to decide if X^2 is in model

```
PROC LOGISTIC DATA = Analysis desc; /* Fit Apparent Model */
CLASS C1 C2;
MODEL Y = X X*X X1-X5 C1-C2 / SELECTION=BACKWARD SLS=.05 HIERARCHY=single;
```

Next, include this same PROC LOGISTIC code into the bootstrap sample model fitting of optimism correction … see next slide …

Handling the "X^2 decision" within optimism correction.

The code below is in macro loop where &I is bootstrap sample number 1 to 200

```
PROC LOGISTIC DATA = BootSamples(where=(replicate=&I)) desc OUTMODEL=OM;
FREQ NumberHits; /* This is bootstrap sample FREQ variable */
CLASS C1 C2;
MODEL Y= X X*X X1-X5 C1-C2 / SELECTION=BACKWARD SLS=.05 HIERARCHY=single;
/* More SAS code follows */
```

For some of the 200 models X^2 is added, otherwise not

… Likewise, sometimes X1, sometimes not

NOTE: Bootstrap models are saved by OUTMODEL=OM

The Optimism Correction process has been preserved !!

Now, continue with the steps to compute the Optimism Correction Lift Chart

NEXT

Optimism Correction … might seem like smoke and mirrors?

- Optimism correction was developed by the renowned statistician Bradley Efron during the period of late 1970's to the 1980's.

  See: Efron and Tibshirani (1993) *An Introduction to the Bootstrap*, pp 247-252

- Frank Harrell, Ewout Steyerberg, and others are leaders in promoting this approach.

  See: Harrell, F. (2015) *Regression Modeling Strategies: With Applications … , 2nd Ed*,
  See: Steyerberg, E. (2019). *Clinical Prediction Models 2nd Ed*.

Strong theoretical and academic support for optimism correction.

Empirical Support for Optimism Correction

Next slides summarize a study by Austin and Steyerberg (2017) (to be called: A-S).

Austin, P. and Steyerberg, E. (2017). Events per predictor (EPP) and the relative performance of different strategies for estimating the out-of-sample validity of logistic regression models, *Stat Methods Med Res.*

The A-S study involves EPP = events per predictor.

Let n1 = events and M = candidate predictors (to be considered for model)
EPP = n1 / M ... If 500 events and 20 predictors, then EPP = 500/20 = 25.

A-S studies relationship of EPP, optimism correction, and split-sampling to the c-statistic.

They show the weakness of split-sampling vs. optimism correction.

- - - - -

In my paper I describe my simulation study of optimism correction vs. split-sampling.

I obtained results similar to A-S.

In Appendix C of the paper there is complete SAS code.

What is an acceptable minimum for EPP ?

A-S uses EPP in their Study

We will assume:

number of events n1 ≤ number of non-events n0.

Let M = number of candidate predictors

Events per predictor (EPP) is simply EPP = n1 / M.

For example, if n1 = 500 and M = 20, then EPP = 500/20 = 25.

What is an acceptable minimum for EPP ?

An often cited rule is n1 / M > EPP = 10,

"At least 10 Events per Predictor".

But EPP guidelines are an area of active research. This is a complex subject with no consensus, much less, simple rules.

The higher the EPP, then the more reliable and stable would be the modeling results.

A-S uses c-statistic to compare split-sample v. optimism correction

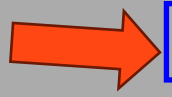
Here is a quick review of the c-statistic

IP ["Informative Pairs"] are pairs of observations (r, s) where Targets $Y_r \neq Y_s$ ... $r \neq s$

Count of IP = (Count of cases where Y=1) * (Count of cases where Y=0)

... here: 3 * 2 = 6

Case ID	Y	P
#1	0	0.1
#2	0	0.2
#3	1	0.1
#4	1	0.2
#5	1	0.3



IP's	Y_i vs. Y_j	P_i vs. P_j
(#1, #3)	$Y_1 < Y_3$	$P_1 = P_3$ ... T
(#1, #4)	$Y_1 < Y_4$	$P_1 < P_4$ ... C
(#1, #5)	$Y_1 < Y_5$	$P_1 < P_5$ ... C
(#2, #3)	$Y_2 < Y_3$	$P_2 > P_3$ ... D
(#2, #4)	$Y_2 < Y_4$	$P_2 = P_4$ ... T
(#2, #5)	$Y_2 < Y_5$	$P_2 < P_5$ ... C

T = Tie
C = Concordant
D = Discordant

c-statistic = (C + 0.5*T) / IP
IP = 6
C = 3 (concordant)
T = 2 (tie)
c-statistic = (3 + 0.5*2) / 6
= 0.667

(#1, #4) ... (Y=0, P=0.1) and (Y=1, P=0.2) ... both Y and P increase
So, (#1, #4) is concordant C.

(#1, #3) ... (Y=0, P=0.1) and (Y=1, P=0.1) ... P's are tied
So, (#1, #3) is tie T

When does the c-statistic say the model is good?

Closer c-statistic is to **1** (… or 0 … we'll focus on 1), then the better is the logistic model

- If c-stat is high, then "generally", more "P" is given to the event than to non-event
- This says: The Model discriminates well between events and non-events

c-stat -- Guidelines

0.5 No discrimination (#C's=#D's)
0.5 < c < 0.7 Poor
0.7 ≤ c < 0.8 Acceptable
0.8 ≤ c < 0.9 Excellent
c ≥ 0.9 Outstanding

See Hosmer et al, (2013 p. 177)

c-stat of at least 0.7 is *adequate.*
Siddiqi: Intelligent Credit Scoring (2017)

Hosmer D., Lemeshow S., and Sturdivant R. (2013). *Applied Logistic Regression*, 3rd Ed.

NEXT

The A-S Study Design … slide 1

A-S used data from Enhanced Feedback for Effective Cardiac Treatment Study with 16,237 patients for their study of optimism correction.

These 16,237 patients will be called the Master File.
Response variable was binary with event-rate of **0.319**.
There were 10 candidate predictors (M=10).

EPP levels of 5 to 100 in increments of 5 (5, 10, 15, …, 100) were studied.

For each EPP level, random samples from the Master File were taken (replacing each sample after selection) until there were 1000 samples. (1000 at EPP=5, 1000 at EPP=10, etc.)

Including all 10 predictors, logistic models were fit to these samples.

The A-S Study Design … slide 2

Average c-statistics (across 1000 samples) were computed for each EPP level for 4 scenarios below.

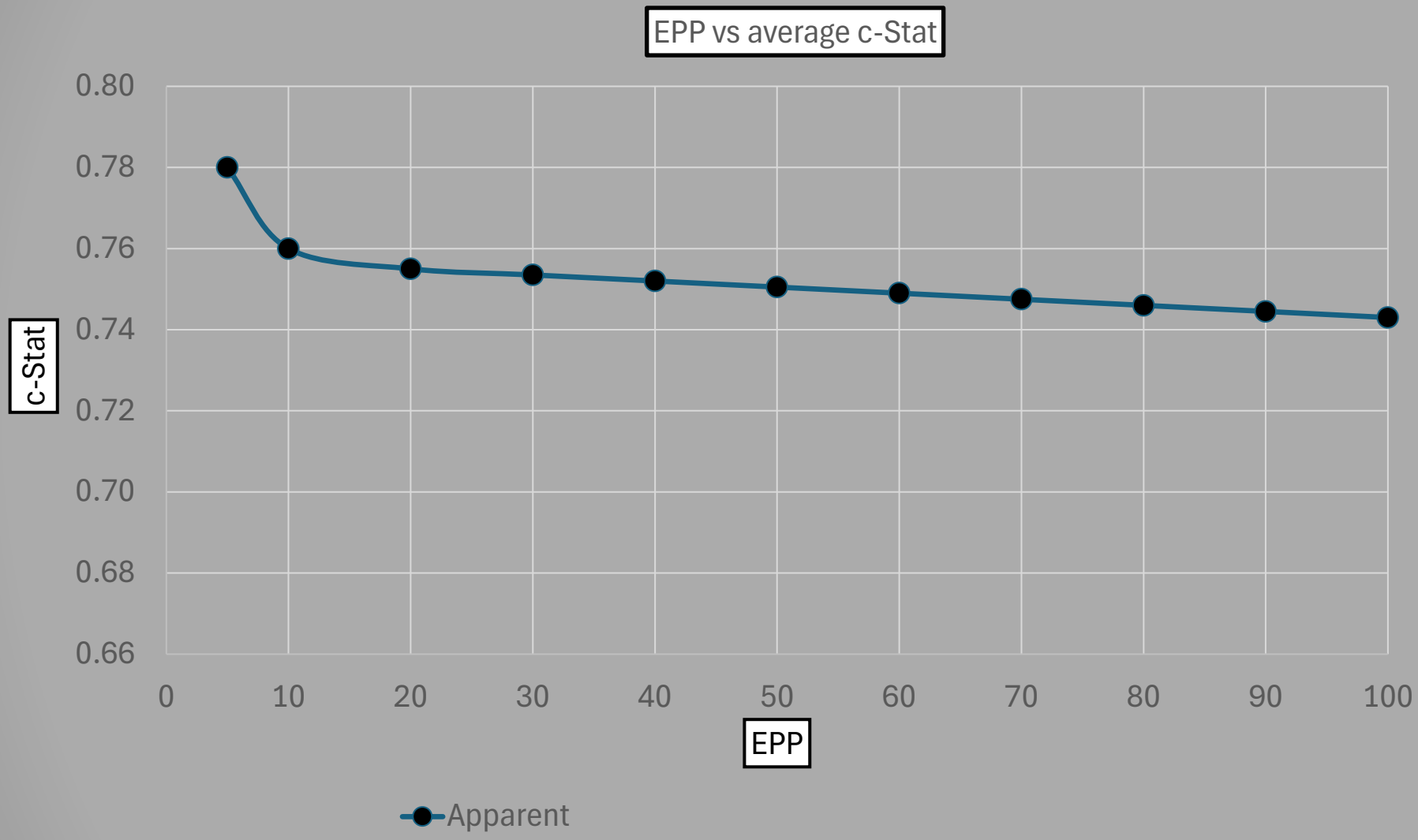
- Apparent: Model fit to sample and compute c-statistic and then take average / 1000.
- External Validation: Model fit to sample and compute c-statistic on rest of Master File. Then take average / 1000.
  This is best approximation to "correct" c-statistic for the Apparent Model (at each EPP).

Of course, in practice, there is no such master file and therefore, no external validation

- Split-Sample (50-50): Model fit to Train and compute c-statistic on Validation and then take average / 1000
- Optimism Correction: Correct the c-statistic from Apparent Model with 100 (not 200) bootstrap samples and then take average / 1000

NEXT

Apparent Model c-statistics



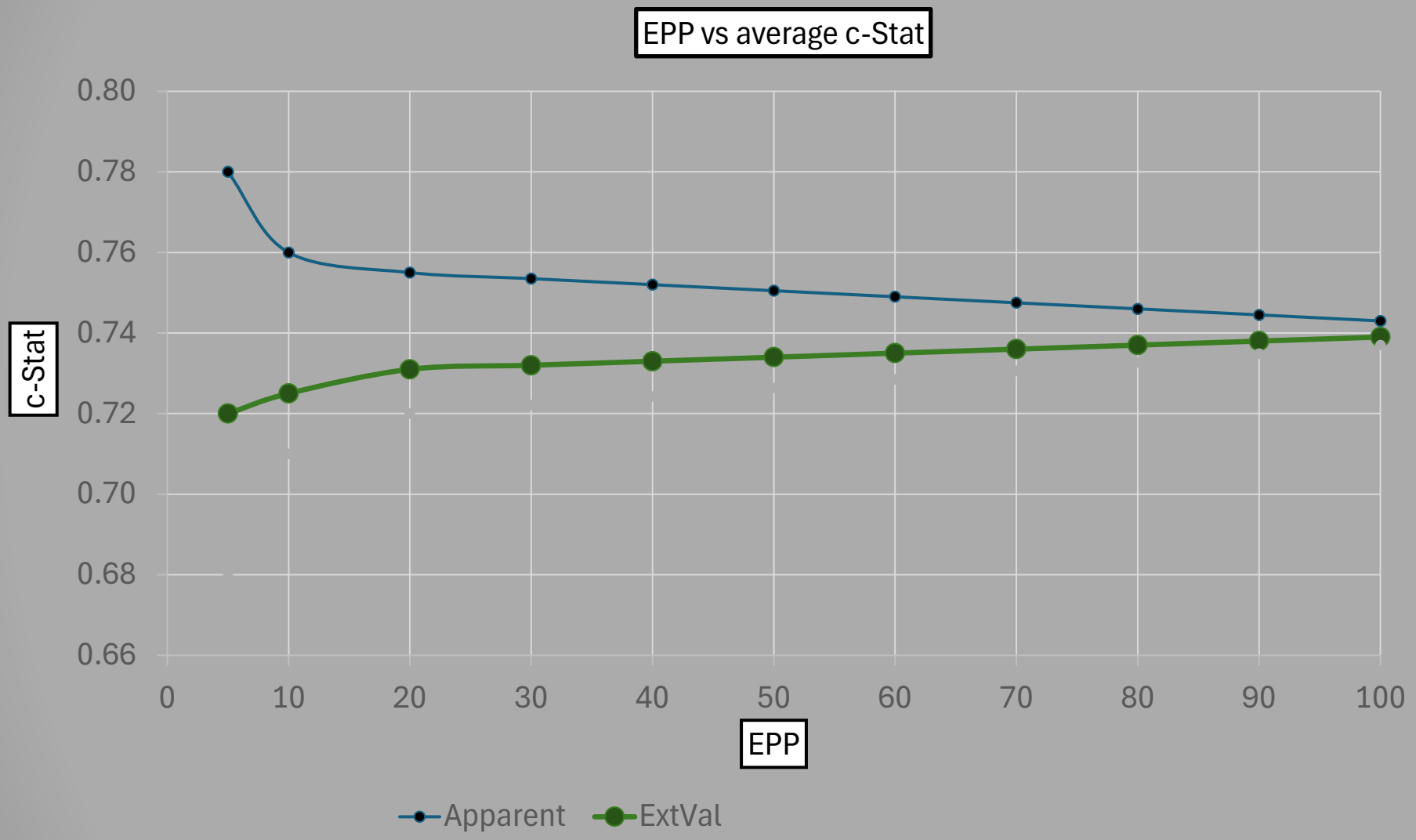
Apparent Model

- 1000 samples per EPP
- Model fit to sample
- c-statistic computed on that sample and averaged (/1000)

c-statistic decreases with higher EPP.

These results pertain to averages over 1000 iterations at each EPP.
Any given iteration could give a wide deviation from the averages.

Add External Validation c-statistics

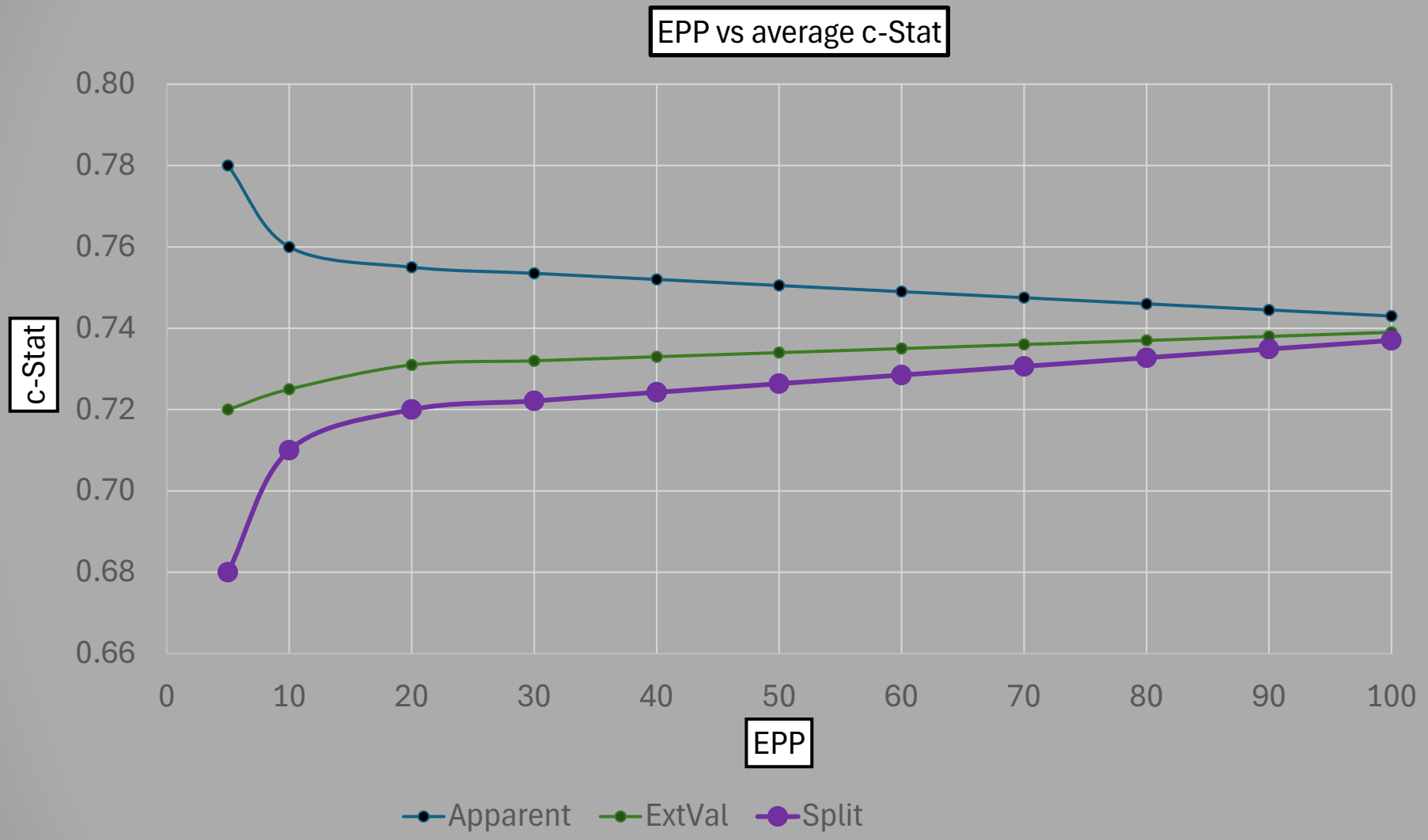


External Validation

- 1000 samples per EPP
- Model fit to sample
- c-statistic computed on rest of Master File and averaged (/1000)
- This is the "correct" c-statistic for the Apparent Model at each EPP.
- Apparent model c-statistic biased high v. External Validation
- External Validation c-statistic increases with higher EPP.

In practice, there is no master file and no such external validation

Add Split-Sample c-statistics

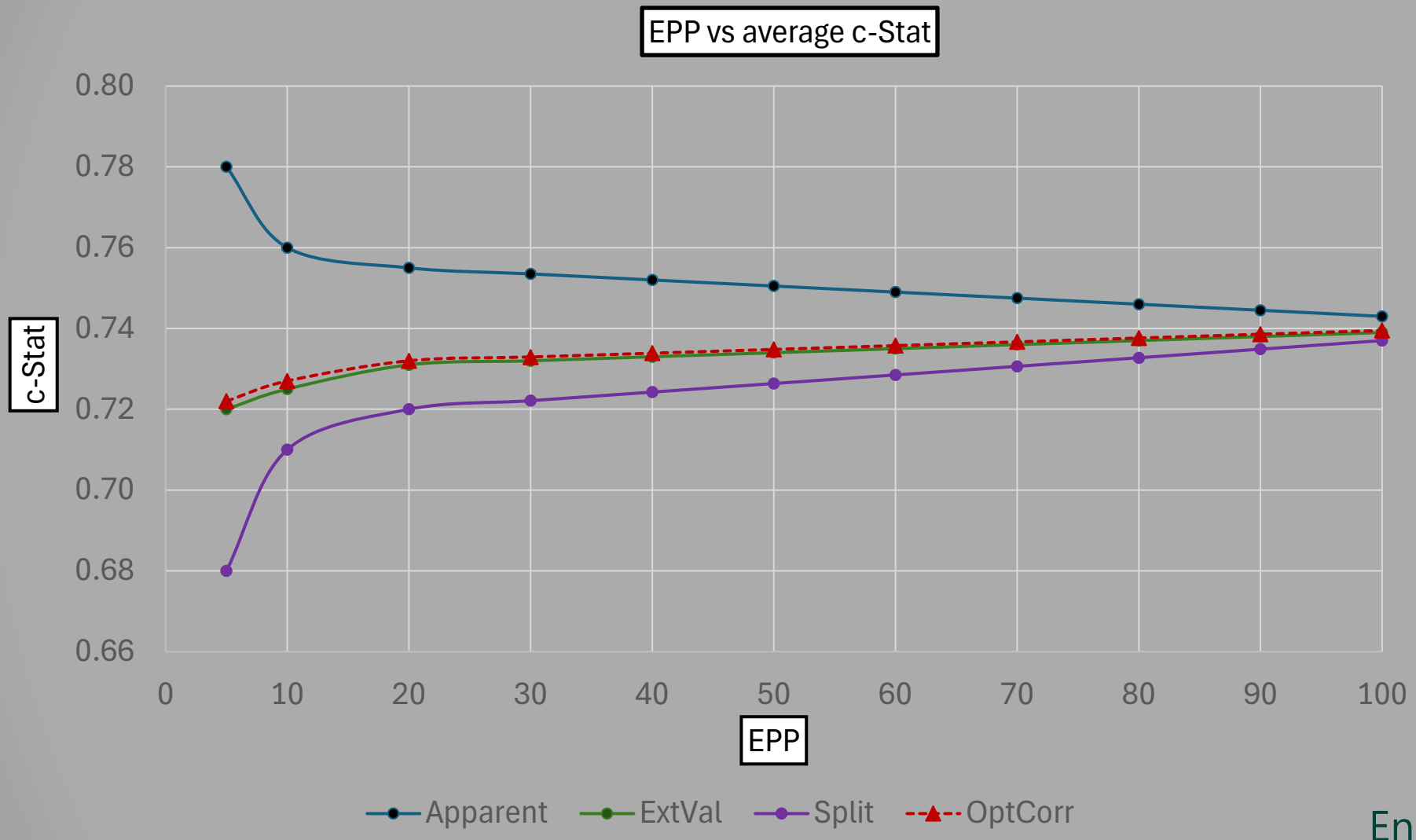


Split Sample

- 1000 samples per EPP
- Model fit on Train (50% of sample)
- c-statistic computed on Validation (50%) and averaged (/1000)
- Split-sample c-statistic much below External Validation.

These results pertain to averages over 1000 iterations at each EPP.
Any given iteration could give a wide deviation from the averages.

Add Optimism Correction c-statistics



Optimism Correction

a) 1000 samples per EPP

b) Model fit to 100 bootstrap samples and c-statistic

c) Bootstrap models score 100 full samples and c-stat computed

d) Subtract (c) from (b) and take the average

OptCorr equals:

Apparent c-statistic – (d)

➔ **OptCorr ~ ExtVal**

End of discussion of A-S

These results pertain to averages over 1000 iterations at each EPP. Any given iteration could give a wide deviation from the averages.

Software

Software:

- I'm not aware of a SAS provided MACRO for optimism correction at this time (but check this!).
- A SAS Global Forum paper gives an example of Optimism Correction and a SAS Macro. See "Estimating Harrell's Optimism on Predictive Indices Using Bootstrap Samples" by I. Stijacic Cenzer, Y. Miao, K. Kirby, W. J. Boscardin (2013).
- The recent book *Building Regression Models with SAS®*, by Robert Rodriguez (2023) includes a discussion of optimism correction for the logistic model (pp. 172-174). SAS code is provided via the authors website. This code computes the optimism correction for the misclassification rate. https://support.sas.com/downloads/package.htm?pid=2688
- Ask ChatGPT to provide SAS code. This code can be adapted to compute optimism correction.

Other Methods to Measure Model Performance … not discussed today

Methods not discussed today that can apply to c-statistic, lift charts, and other metrics:

- .632 bootstrap estimator
- .632+ bootstrap estimator
- Cross-validation methods

How to Learn about methods to measure logistic model (and other model) performance:

- Ask ChatGPT or other AI for discussion of these methods (I found this to be very helpful)
- Google Search for your specific interests or needs (videos, short discussions, etc.)
- Efron and Tibshirani (1993) *An Introduction to the Bootstrap*, See Chapter 17
- Trevor, Tibshirani, and Friedman (2009) *The Elements of Statistical Learning*
  See 7.10 Cross-Validation and 7.11 Bootstrap Methods

Is this the End of Split-Sampling?

NEXT

For analysis dataset with N ~ 1000 and event-rate ~ 0.5, a split sample seems inappropriate. (Recall the 20 split-samples example.)

What about an analysis dataset with N ~ 20000 and large EPP (say > 50)?

- It is never wrong to use Optimism Correction (provided no double dipping)
- However, a split sample for N ~ 20000 and EPP > 50 is probably OK
- Split-sampling allows exploration of predictors vs. target in a free flowing manner and, as well, the opportunity to try various modeling techniques. The Validation sample gives the final unbiased model assessment.

If N ~ 1000 and event-rate ~ 0.5, what about "data exploration" … looking at X's vs. Y? This is a normal pre-modeling process which gives a modeler a feel for the data and for potential modeling approaches. Is this double dipping?

YES !! … if using optimism correction.

To avoid: Incorporate variable transformation and variable selection steps into the optimism correction process.

Logistic Modeling without Split-Samples

BRUCE LUND
Statistical Trainer, Novi, MI

See my Paper in MISUG.org for SAS code

Blundz@iCloud.com … or
Blund.data@gmail.com

v2

Appendix A: Cross-Validation for “CHOOSING” the model

The next slides show how Cross Validation can be used to “CHOOSE” a model during variable selection and model fitting.

The first slide shows how PROC GLMSELECT uses Cross Validation “CHOOSE” for ordinary linear regression. This feature is only available in PROC GLMSELECT.

Then, in the next four slides, I show how Cross Validation “CHOOSE” might be applied to logistic regression. But the computational requirements would be significant. I don’t think my method is practical for large models.

PROC GLMSELECT applied to Y, X1, X2 from WORK

```
PROC GLMSELECT DATA=WORK;
MODEL Y = X1-X2 / CVMETHOD=SPLIT(5)
CVDETAILS=ALL
SELECTION=FORWARD
(SELECT = AIC CHOOSE = CV STOP=NONE);
run;
```

← Add predictors according to AIC and CHOOSE the MODEL with minimum average, across the 5 test datasets, of the ASE

Forward Selection Summary				
Step	Effect Entered	Number Effects In	AIC	choose= CV PRESS
* Optimal Value of Criterion				
0	Intercept	1	-382.2944	250.0000
1	X1	2	-407.0926	243.5871
2	X2	3	-425.1889*	239.1676*

Cross Validation Details			
Index	Observations		CV PRESS
	Fitted	Left Out	
1	800	200	48.2089
2	800	200	47.2334
3	800	200	47.9582
4	800	200	48.2786
5	800	200	47.4885
Total			239.1676

CV_PRESS = $\sum_{i=1}^{5} \sum (Y - \hat{Y})^2$ for Y's from sum of holdouts

Divide by 5 and then by 200 to obtain 0.239168
This is ~ASE as a later an adaptation of HPLOGISTIC

Of course, target is BINARY – a Violation. Still, this approach is OK, at least as a first approximation.

"CHOOSE=CV using HPLOGISTIC" … copies idea from GLMSELECT

STEP 1

```
DATA WORK; SET German.Bank_X_Coded;
PROC HPLOGISTIC DATA = WORK;
MODEL Y (descending) = X1 X2;
SELECTION METHOD=FORWARD (SELECT=AIC CHOOSE=AIC STOP=NONE)
DETAILS=ALL;
run;
```

Selection Summary

Step	Effect Entered	Number Effects In	AIC
0	Intercept	1	1388.29
1	X1	2	1363.85
2	X2	3	1345.57*

* Optimal Value of Criterion

"CHOOSE=CV using HPLOGISTIC" … copies idea from GLMSELECT

STEP 2: Divide WORK into 5 random samples (this is just 1 "repeat"):

```
DATA WORK2; SET WORK; random = ranuni(1);
PROC SORT DATA = WORK2; BY random;
DATA WORK2; SET WORK2; DROP I;
ARRAY ROLES (5) ROLE1-ROLE5;
DO I = 1 to 5;
IF MOD(_N_-1,5)+1 = I THEN ROLES(I) = 1;
ELSE ROLES(I) = 0;
END; /* Or better, use SURVEYSELECT */
run;
```

"CHOOSE=CV using HPLOGISTIC" … copies idea from GLMSELECT

STEPS 3a for X1 (1st X in ordering)

```
PROC HPLOGISTIC DATA = WORK2;
PARTITION ROLEVAR= ROLE1
(TRAIN="0" VALIDATE="1");
MODEL Y (descending) = X1;
PROC HPLOGISTIC DATA = WORK2;
PARTITION ROLEVAR= ROLE2
(TRAIN="0" VALIDATE="1");
MODEL Y (descending) = X1;
PROC HPLOGISTIC DATA = WORK2;
PARTITION ROLEVAR= ROLE3
(TRAIN="0" VALIDATE="1");
MODEL Y (descending) = X1;
PROC HPLOGISTIC DATA = WORK2;
PARTITION ROLEVAR= ROLE4
(TRAIN="0" VALIDATE="1");
MODEL Y (descending) X1;
PROC HPLOGISTIC DATA = WORK2;
PARTITION ROLEVAR= ROLE5
(TRAIN="0" VALIDATE="1");
MODEL Y (descending) = X1;
run;
```

ASE X1
0.2408
0.2477
0.2476
0.2448
0.2410
ASE Average
0.2444

"CHOOSE=CV using HPLOGISTIC" … copies idea from GLMSELECT

Selection Summary			
Step	Effect Entered	Number Effects In	AIC
0	Intercept	1	1388.29
1	X1	2	1363.85
2	X2	3	1345.57*

* Optimal Value of Criterion

STEP 3b REPEAT Step 3 by adding X2 to X1 to the MODEL statement

Find ASE Average for 5 x-folds

ASE steps 3ab	
X1	0.2444
X1 X2	0.2399 (*)

STEP 4 … **CHOOSE** the MODEL with min. average ASE

This model is ➔

But a single 5-fold or 10-fold cross-validation, some experts say, is not nearly enough repetitions.

Frank Harrell writes: "Need to do 50 repeats of 10-fold cross-validation to ensure adequate precision." See https://hbiostat.org/rmsc/ section 5.3.4

Appendix B: Formula for the number of ranks for a Lift Chart

The next slides show the formula for obtaining the number of ranks for a lift chart that might lead to a *consistent* lift chart. First, definition of quantities:

Y_mean =	The event-rate for the dataset.
Freq =	Number of observations in the dataset.
SD =	Sqrt(Y_mean*(1-Y_mean)/Freq)
Z_value =	1.65 where 1.65 is chosen for the pseudo 90% confidence interval (CI) shown below for the "true" population Y_mean: CI = (-1.65 * SD + Y_mean, +1.65 * SD + Y_mean)
Freq_rank =	Number of observations within a rank (it is assumed that Freq_rank applies to all ranks).
SD_rank =	Sqrt(Y_mean*(1-Y_mean)/Freq_rank)
P_range =	P99 – P01 where P99 and P01 where P99 is the 99th percentile of P for the fitted model and P01 is the 1st percentile of P for the fitted model. (These percentiles are across ALL observations.)
K =	Number of ranks for the Lift Chart (to be solved for)

Here is the formula for K:

$$K = (P\_range / (2 * Z\_value * SD))^{2/3}$$

Derivation is on the next slide

Derivation

Derivation

This constraint determines K. The K summed lengths of the 90% CI's must equal P_range.

P_range = K * 2 * Z_value * SD_rank … (A)

An identity for K is given by:

Freq_rank = Freq / K … (B) … (note: Freq_rank is unknown):

Substituting the expression for Freq_rank from (B) into (A) gives:

P_range = K * 2 * Z_value * sqrt[Y_mean*(1-Y_mean) * K / Freq]

P_range = K * 2 * Z_value * SD * sqrt(K)

Solving for K:

$K = (P\_range / (2 * Z\_value * SD))^{2/3}$

Examples:

If: P_range = 0.8	Z_value = 1.65	Y_mean = 0.5	Freq = 5000,	then K = 10.55
If: P_range = 0.8	Z_value = 1.65	Y_mean = 0.48	Freq = 500,	then K = 4.90
If: P_range = 0.65	Z_value = 1.65	Y_mean = 0.1	Freq = 10000,	then K = 16.27
If: P_range = 0.99	Z_value = 1.65	Y_mean = 0.1	Freq = 10000,	then K = 21.54

But the formula is not guaranteed to give a consistent Lift Chart or, otherwise, may not give the maximum number of ranks that give a consistent Lift Chart. But it gets the user "into the ballpark" for finding a good value of K.

An Example

Inputs for calculation for the Model on fitted on Train

Obs	Freq	P01	P99	Y_mean
ALL	500	0.270	0.733	0.5

where P99 and P01 are the 99th and 1st percentiles of P

Here is the formula for K, the number of ranks.

$$K = (P\_Range / (2 * 1.65 * SD))^{2/3}$$

P_Range = P99 – P01 where P99 and P01 are the 99th and 1st percentiles of P
SD = Sqrt(Y_mean * (1-Y_mean) / Freq) … see Freq and Y_mean in Table
1.65 = Is half the width of 90% (pseudo) Confidence Intervals for "true" Y_mean

Rationale: Pick K so that P_Range = K * width of 90% Conf. Int. = K * (2 * 1.65 * SD)

$K = (0.463 / (2 * 1.65 * 0.022361))^{2/3} = 3.40$ … and round up to **4 ranks**