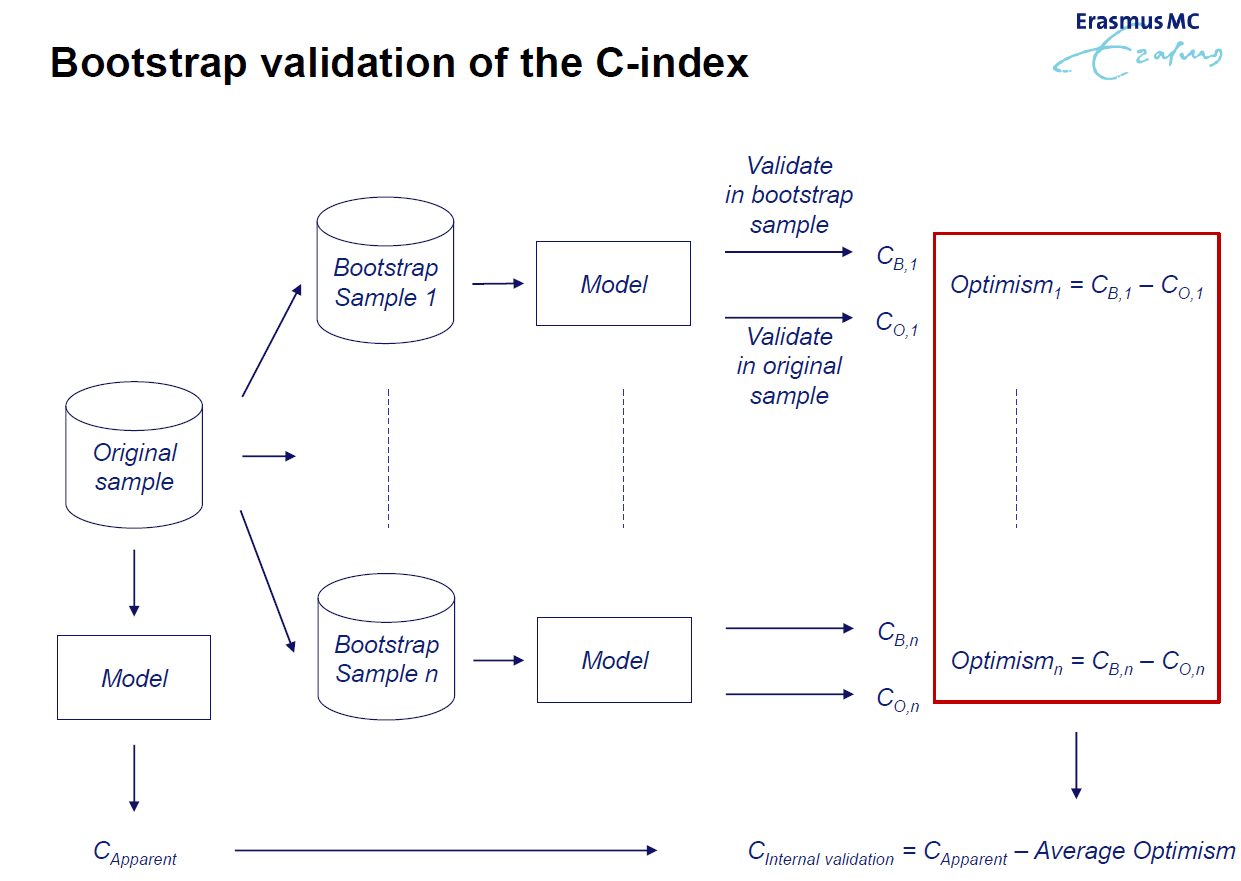
**HYPOPARATHYROIDISM**

**Final model**

Fit final model (all possible predictors) on original data set. Perform backward selection to obtain the final model with only those predictors that are strongly predictive. This gives original coefficients (intercept , coefficients of predictors ) + odds ratio’s (OR = ) + 95% confidence intervals for the final model.

**Shrunk coefficients**



1. Create a bootstrapped sample of the original sample
2. Perform backward selection of the full model (all possible predictors) each of the bootstrapped sample
3. Calculate the calibration slope on the original set using the model fit on each of the bootstrap sample ( in bovenstaande figuur)
4. Shrinkage factor is the calibration slope of the models fitted on the bootstrapped sample and tested on the original data set, averaged over the ten imputed data sets

*Shrunk coefficients of the predictors* are and the shrunk odds ratio’s are . The confidence intervals of the shrunk coefficients are not informative, since multiplying the coefficients with a constant factor will not change the width of the interval, it would only shift the confidence interval left or right. Of course we can calculate how much to shift the interval, but it is a bit weird. Therefore, I excluded the confidence intervals from the webapp.

*Shrunk intercept* is . Since we are shrinking the coefficients we need to add a constant () to predict on average the event probability.

*Calculate .* Fit on the offset of , where the outcome, the shrinkage factor, and the linear predictor. The offset ensures the coefficient for is set to 1. The intercept of this regression lrm.fit(y, offset=lambda\*LP) is .

**READMISSION**

**Calibrated coefficients**

We have a model that predicts hypoparathyroidism pretty well, and we want to use **the same** model to predict readmission. However, readmission occurs less frequently, so we want to calibrate the model to predict readmission outcome by multiplying the shrunk coefficients of the hypoparathyroidism model with a calibration factor .

*The calibrated shrunk coefficients* are , where the calibration factor is obtained by regressing the new outcome readmission on the linear predictor of the final model without shrinkage, averaged over the ten imputed data sets (lrm(Readmission~LP)). The calibration factor is the coefficient of the linear predictor obtained using the final hypoparathyroidism model.

*The calibrated shrunk intercept* are , where is the intercept of the logistic regression of readmission on the offset of the calibrated shrunk linear predictor, e.g. lrm.fit(Readmission, offset=gamma\*lambda\*LP).

**Refitted coefficients**

Instead of using one factor to multiply all coefficients, we could also use the same predictors chosen by backward selection for hypoparathyroidism, and refit the coefficients to tailor the model to predict readmission. This is what we call “refitting”, but as you can see the discriminative ability is not much better than simply calibrating coefficients.

I’ve shown the refitted coefficients without shrinkage, because if you want to apply shrinkage, you’d have to do backward selection to obtain the calibration slope through backward selection, which amounts to “redeveloping” the model. The same holds for correcting for optimism, which also involves bootstrapping and backward selection. I’m only showing the refitted coefficients to show that they are not much different than the coefficients (without shrinkage) for the hypoparathyroidism model, and that the C-index is not that different as well.

**Redevelop model**

To redevelop the model is to pretend there is no model for hypoparathyroidism, and apply backward selection on the full model to see which predictors would be relevant to predict readmission. In our case that is only PTH and corrected calcium at 24 hours. Thus, BSKgezien is not predictive for readmission. I didn’t show this model in the results, because we want one model with the same predictors to predict two outcomes. This way, we have one webapp to display. Furthermore, the model doesn’t perform much better when you exclude BSKgezien.