Obstacles and pitfalls

for the prediction modeling strategy in general and the estimation of prediction performance in particular

Outline

Advanced statistical topics arise routinely in the prediction modeling process and require special attention.

Many issues have no clear solutions but they are still important to recognize.

However, even unsolved, these issues are important to point out when discussing the limitations of a study, and potentially could motivate the target of a sensitivity analysis.

Cross-validation, Censored outcome, Missing predictor values

Cross-validation

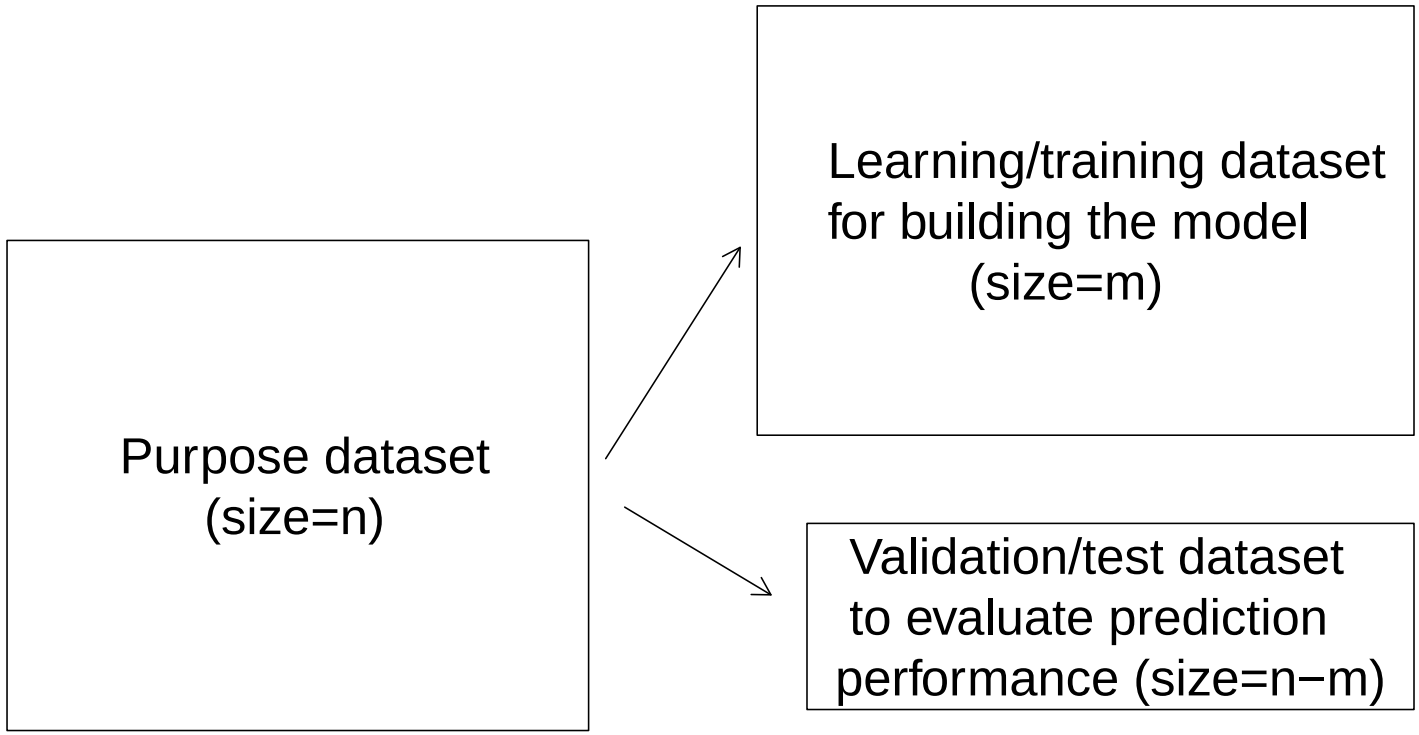
Data splitting

Internal cross-validation can be used to tune model parameters and to assess and compare the predictive performance of a list of candidate models (AKA super learner).

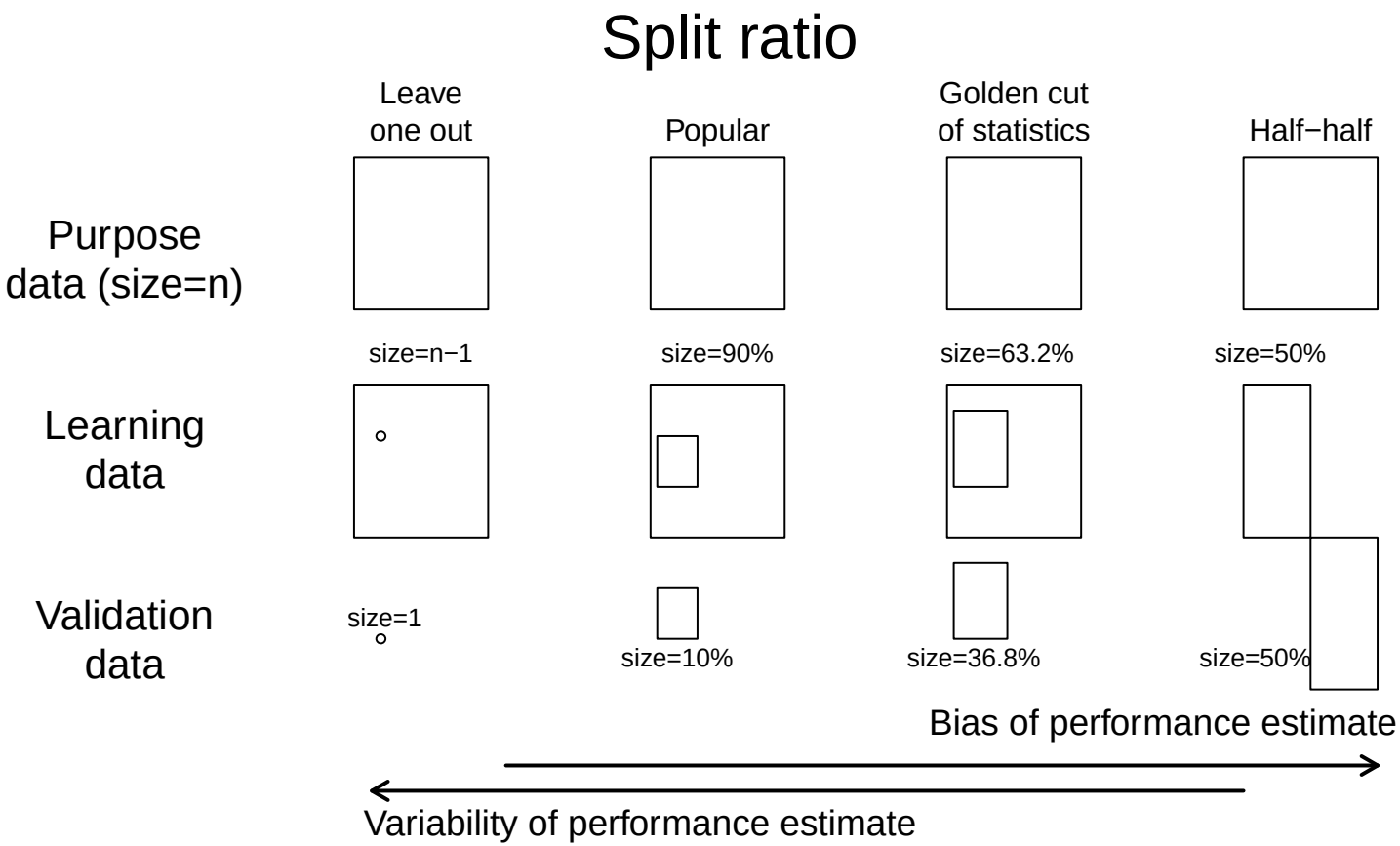
"Internal" can either mean that the "validation" study is performed by the same team who made the model, or characterize a "validation" study which splits a purpose data into training and validation sets.

We generally discourage the use of a single split of data into one set for training and one for validation.

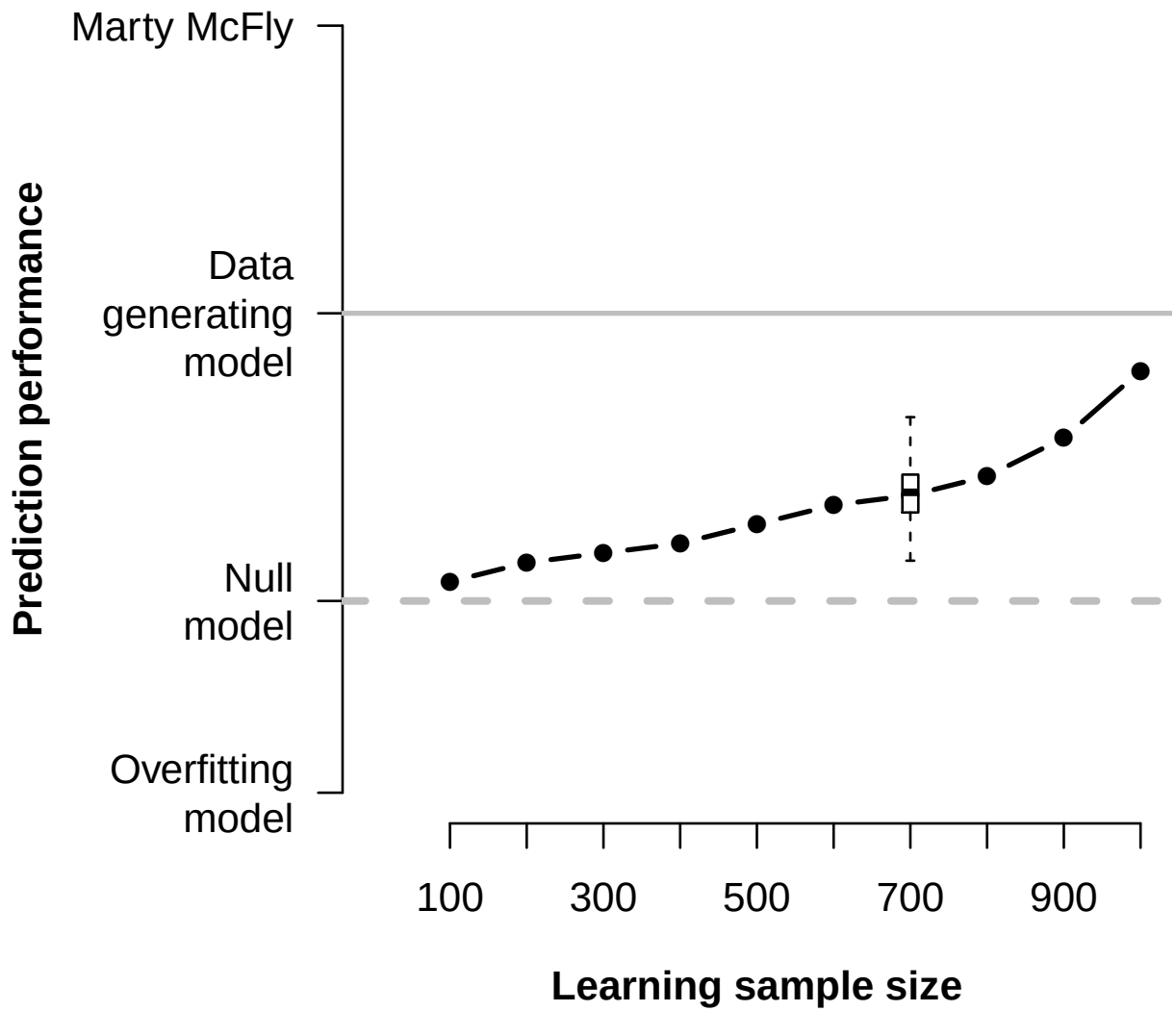
Data splitting



Data splitting



Learning curve



Example: single split results depend on seed

```
x <- Score(list("Conventional model" = conventional_
    model,"Experimental model" = experimental_model),
           data = train,
           formula = cvd_5year~1,
           split.method = "bootcv",
           progress.bar = NULL, verbose = FALSE,
           null.model = FALSE, se = FALSE,
           seed = 17,
           B = 1,
           M = .632*NROW(train))
summary(x,what = "score")
```

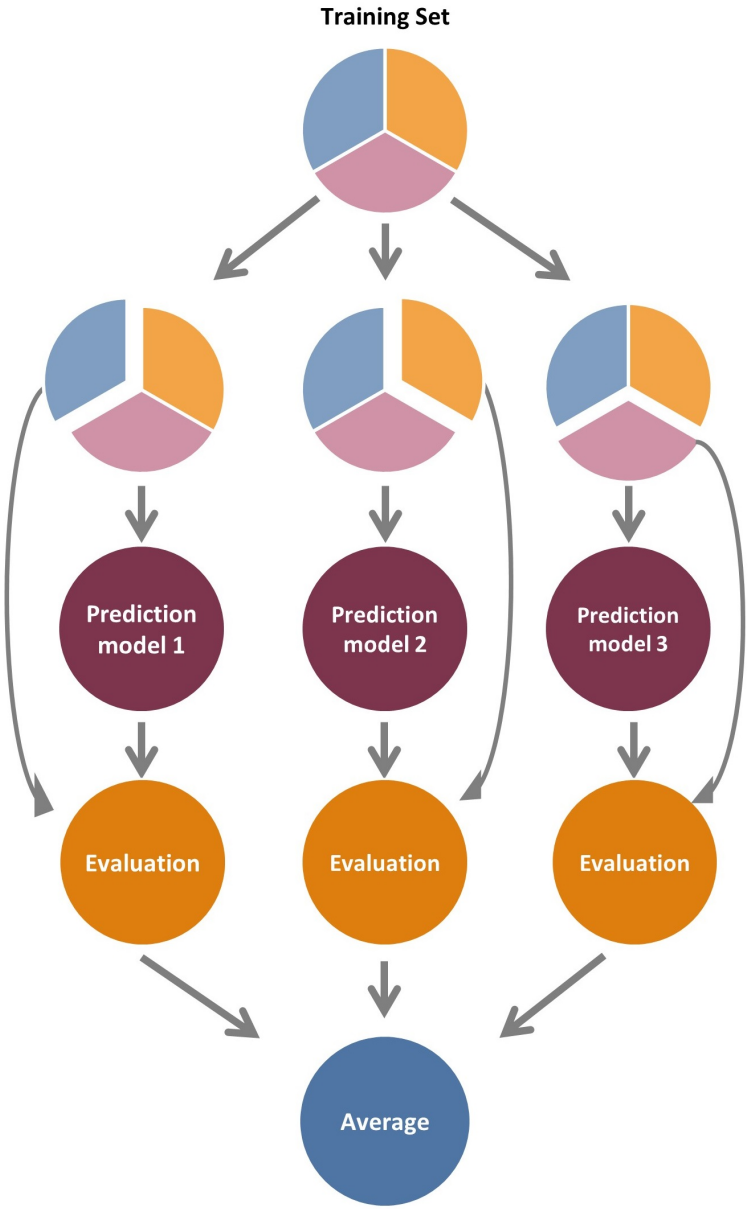
```
$score
Key: <Model>
                  Model AUC (%) Brier (%)
                 <fctr>  <char>    <char>
1: Conventional model    91.6       5.7
2: Experimental model    90.3       5.9
```

Example: single split results depend on seed

```
x <- Score(list("Conventional model" = conventional_
    model,"Experimental model" = experimental_model),
           data = train,
           formula = cvd_5year~1,
           split.method = "bootcv",
           progress.bar = NULL, verbose = FALSE,
           null.model = FALSE, se = FALSE,
           seed = 4,
           B = 1,
           M = .632*NROW(train))
summary(x,what = "score")
```

```
$score
Key: <Model>
                  Model AUC (%) Brier (%)
                 <fctr>  <char>    <char>
1: Conventional model    93.5       5.7
2: Experimental model    93.6       5.6
```

Repeated cross-validation helps (but changes the target!)



Repeated cross-validation helps (but changes the target!)

```
x <- Score(list("Conventional model" = conventional_
    model,"Experimental model" = experimental_model),
           data = train,
           formula = cvd_5year~1,
           progress.bar = NULL, verbose = FALSE,
           null.model = FALSE, se = FALSE,
           split.method = "bootcv",
           seed = 17,
           B = 100,
           M = .632*NROW(train))
summary(x,what = "score",digits = 2)
```

```
$score
Key: <Model>
                  Model AUC (%) Brier (%)
                 <fctr>  <char>    <char>
1: Conventional model    90.95      4.98
2: Experimental model    90.62      5.11
```

Repeated cross-validation helps (but changes the target!)

```
x <- Score(list("Conventional model" = conventional_
    model,"Experimental model" = experimental_model),
           data = train,
           formula = cvd_5year~1,
           progress.bar = NULL, verbose = FALSE,
           null.model = FALSE, se = FALSE,
           split.method = "bootcv",
           seed = 4,
           B = 100,
           M = .632*NROW(train))
summary(x,what = "score",digits = 2)
```

```
$score
Key: <Model>
                  Model AUC (%) Brier (%)
                 <fctr>  <char>    <char>
1: Conventional model    90.88      4.98
2: Experimental model    90.60      5.11
```

Conditional versus expected performance

Conditional prediction performance is the performance of a risk prediction model conditional on a single-purpose dataset.

A researcher who provides a risk prediction model for clinical application is naturally interested in the conditional performance of the model.

The conditional prediction performance can be assessed by an external validation dataset or, with some limitations, using data splitting.

Conditional versus expected performance

Expected prediction performance is the performance of a modeling algorithm. It is the average performance across all the prediction models that a modeling algorithm can produce using all possible learning datasets of a fixed sample size.

A researcher who has invented a new algorithm for building prediction models is naturally interested in the average performance.

The expected prediction performance can either be assessed by computer simulation of many datasets or by using cross-validation and bootstrap methods.

Censored time to event data

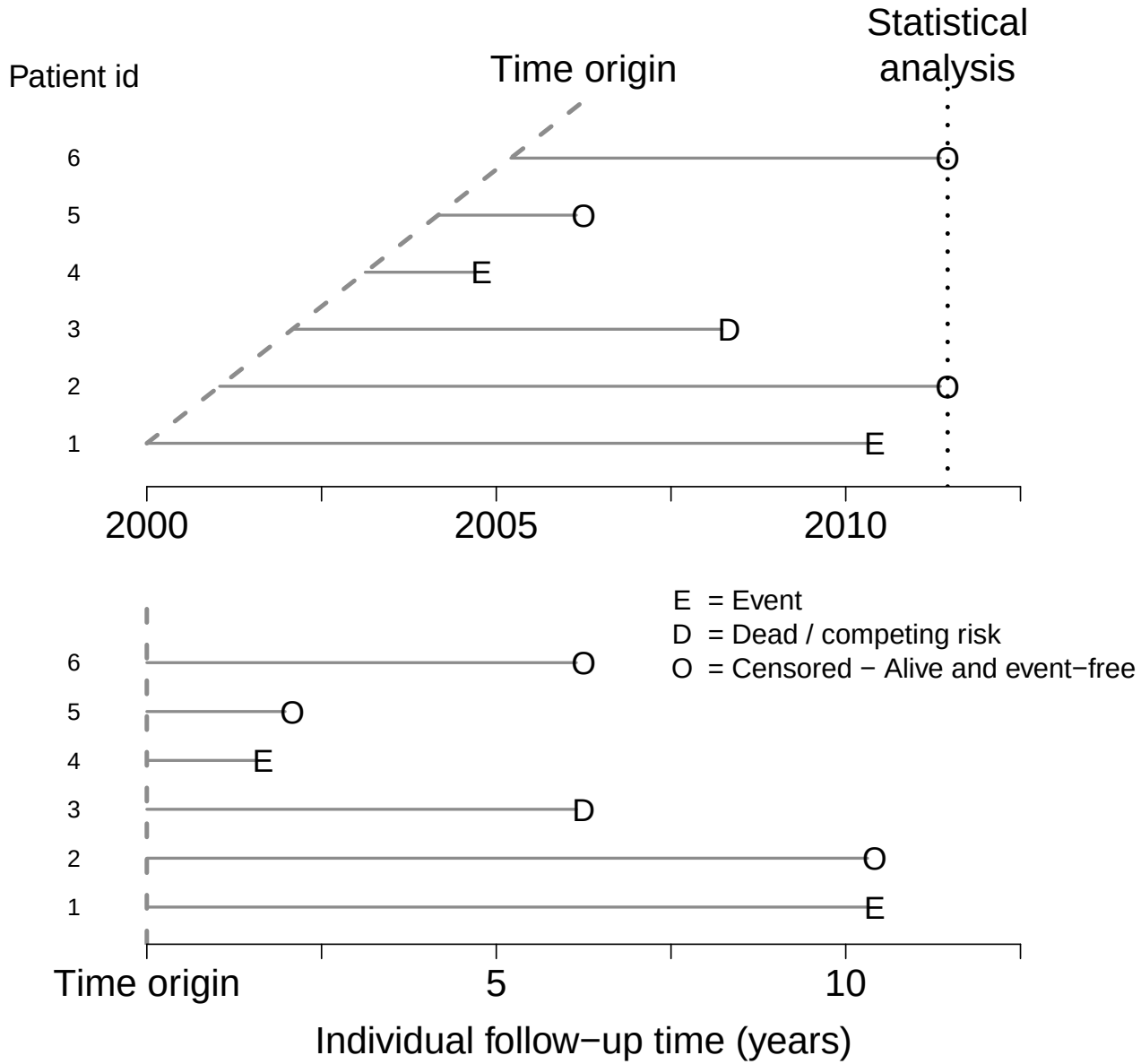
Censored data

A common behavior is that patients with short follow-up are excluded from the dataset. This is almost never the right thing to do, as intuitive as it may seem. Excluding them results in bias.

Outcome at the prediction time horizon is unknown (censored) for subjects who were not followed until the prediction time horizon and were free of any event by the end of their individual follow-up period.

Subjects who die for other reasons before the time horizon are not censored!

Specific estimation techniques are needed to take care of bias due to censored data.



At the 5-year prediction time horizon, only patient 5 is censored. At the 10-year prediction time horizon, patients 5 and 6 are censored.

Inverse probability of censoring weighting

To estimate the average Brier score we calculate a weighted average for only those subjects for who the event status at the prediction time horizon is known (groups *event* and *event-free*).

The weight of a subject is higher the more similar subjects were lost to follow-up (*censored*) earlier in time.

This is to account for the fact that those in the *censored* group could possibly have experienced the event by the prediction time horizon, just this is not observed.

Subjects in the *censored* group are not directly used in the weighted average. However, they enter indirectly through the weights assigned to other subjects.

Choice of the prediction time horizon

The prediction time horizon should be chosen such that it is of subject matter interest to predict the probability that the event occurs in the time period between the time origin and the prediction time horizon.

The shorter the prediction time horizon the fewer subjects are censored!

It is sometimes useful to study multiple prediction time horizons, but this should be motivated by the application at hand (beware of cherry-picking).

Example code (does not work with course data)

```
x <- Score(list("Conventional model" = conventional_
    model,"Experimental model" = experimental_model),
           data = train,
           formula = Hist(time,event)~age+sex+...,
           times = 1:10,
           split.method = "cv10",
           seed = 4,
           B = 100,
           M = .632*NROW(train))
```

Example code (does not work with course data)

```
x <- Score(list("Conventional model" = conventional_
    model,"Experimental model" = experimental_model),
           data = train,
           formula = Hist(time,event)~age+sex+...,
           times = 1:10,
           split.method = "cv10",
           seed = 4,
           B = 100,
           M = .632*NROW(train))
```

The inverse probability of censoring weights are obtained with a regression model for the censoring probabilites based on the variables on the right hand side of the formula. Unfortunately, the censoring regression model can be misspecified ...

Missing predictor values

Missing predictor values

When the target of the analysis is an association parameter such as an odds ratio or a hazard ratio, then *multiple imputation* or *inverse probability weighting* may increase power compared to a *complete case analysis*.

However, it is not so simple that multiple imputation has always less bias than a complete case analysis.

When the aim is to build and "validate" a risk prediction model, we need to deal with missing values in two conceptually different places:

- missing values in the learning dataset
- missing values in data of a new patient who is asking for a predicted risk

Reason for missing predictor values

We need to understand why the values are missing and explore the reasons.

Missing completely at random	The probability of a missing value is not related to either outcome or predictor variables.
Missing at random	The probability of a missing value is predictable by the other predictor variables and outcome.
Missing not at random	The probability of a missing value depends on the missing value and/or on other unobserved variables.

An inconvenient truth to realize is that the type of missingness can generally not be inferred from the data.

Learning phase (Making of a prediction model)

Missing values in the predictor variables affect the choice of predictor variables: having few predictor variables with a high percentage of missingness can be equally bad as having many predictor variables with few missing values in different subjects.

Any modelling algorithm (logistic regression, random forest, etc.) may use missingness of a variable as a predictor and also use imputation of missing values as part of the algorithm!

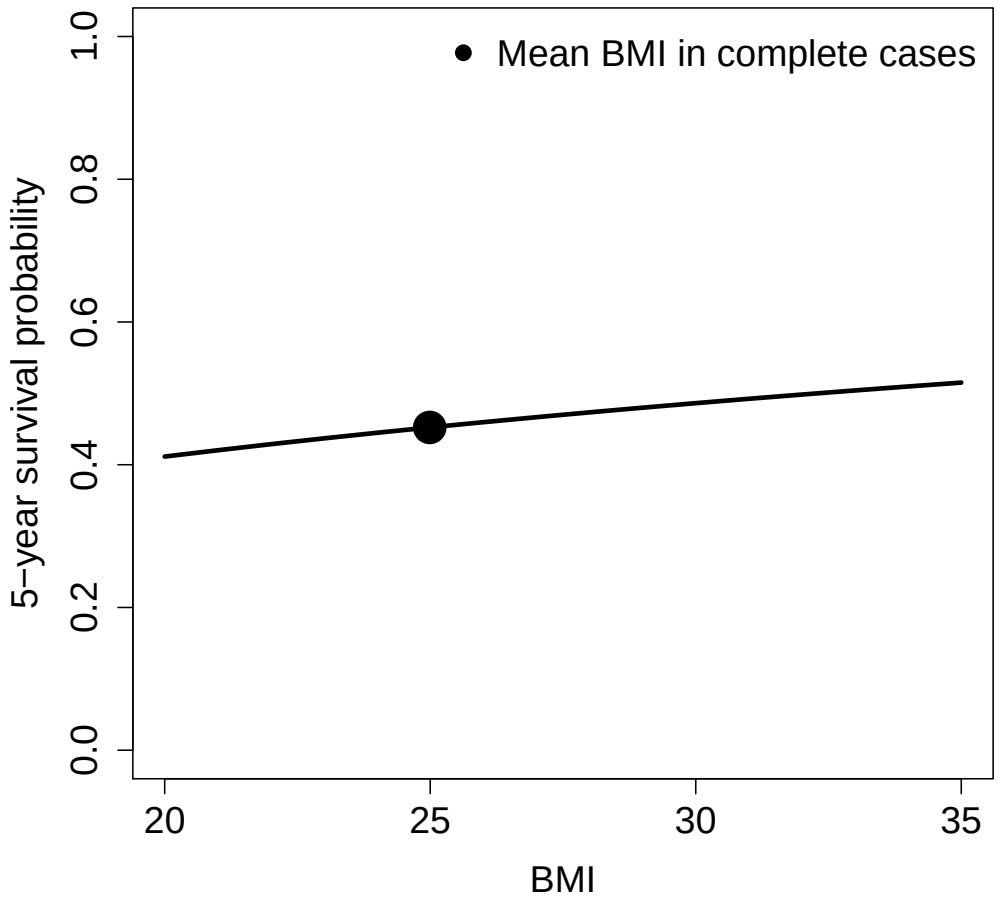
Imputation means to replace a missing value with a likely value. For example, to impute a value for BMI we could specify a linear regression model which relates BMI to the other predictor variables, other auxiliary variables, and the outcome.

Imputation algorithms should also condition on outcome

Thinking about how missingness can possibly depend on an outcome variable is generally cumbersome, and it does not get any easier when the outcome is a right-censored time-to-event variable.

If the missingness of a predictor variable "happens" at baseline, then there cannot be a direct causal effect of the outcome variable on the missingness.

Hence, if the missingness depends on a time-to-event variable conditional on the observed predictor variables, this must mean that there exists an unobserved variable, such as disease burdon, which mediates the relationship.



Suppose the substantive model is a Cox regression model which shows that the 5-year survival probability increases with increasing BMI. Without conditioning on the outcome, we simply impute missing BMI values for every one from a normal distribution with mean given by the mean BMI in the complete cases. When the 5-year survival probability is an increasing function of BMI as in the figure this imputed value (the black dot on the line) is systematically too large for subjects with BMI below 25 and systematically too low for subjects with BMI above 25.

Missing values in the validation data

We distinguish the following two tasks.

The first task is to have the estimator of prediction performance deal with missing values in the validation dataset.

The second task is to enrich the model such that the input values of some of the predictor variables become optional for the user of the model.

That is, if the user of the model cannot provide the value of one or several predictors, ideally, then the model should still be able to provide a predicted risk.

Obviously, all this makes most sense when the missingness in the validation dataset resembles the expected missingness in the future users of the model.