Overview of topics related to model selection for regression

Riccardo De Bin[1]

based on a joint work with Carine Legrand[2], Herbert Braselmann[3,4], Julia Hess[3,4], Anne-Laure Boulesteix[1] and Kristian Unger[3,4]

Barcelona, October 26^{th} 2015

[1]Department of Medical Informatics, Biometry and Epidemiology, University of Munich

[2]Institute of Medical Biometry and Informatics, University of Heidelberg

[3]Research Unit Radiation Cytogenetics, Helmholtz Zentrum München, German Research Center for Environmental Health

[4]Clinical Cooperation Group "Personalized Radiotherapy in Head and Neck Cancer", Helmholtz-Zentrum München/University of Munich

Outline of the talk

- Introduction
- Methods
  - To explain or to predict?
  - Two statistical methods for prediction
  - Combining clinical and molecular data in a prediction model
- Implementation
  - Data
  - Models
- Evaluation
  - Model evaluation
  - Remarks on model evaluation
- Conclusions

Introduction: notation

- regression analysis: study of the relationship between Y and X;
- Y is the outcome (or dependent variable or response variable), which can be:
  - continuous, e.g. *percentage of body fat*, *expiratory volume*, ...
  - binary, e.g. *case/control*, *win/lose*, ...
  - survival time, e.g. *overall survival time*, *time to relapse*, ...
  - ...
- X is the design matrix;
- each column of X is called predictor (or independent variable or covariate). Examples are:
  - clinical predictors, e.g. *age*, *sex*, ...
  - molecular predictors, e.g. *microarray expression*, *copy number alteration*, ...
  - ...

To explain or to predict? Shmueli (2010)

Consider the formal representation

$$E[Y] = f(X).$$

A statistical model can be used:

TO EXPLAIN

- the focus is on the statistical model f;
- the design matrix X and the outcome Y are tools to estimate f.

TO PREDICT

- the focus is on the design matrix X and the outcome Y;
- the statistical model f is the tool to predict Y given X.

To explain or to predict? Bias-variance trade-off

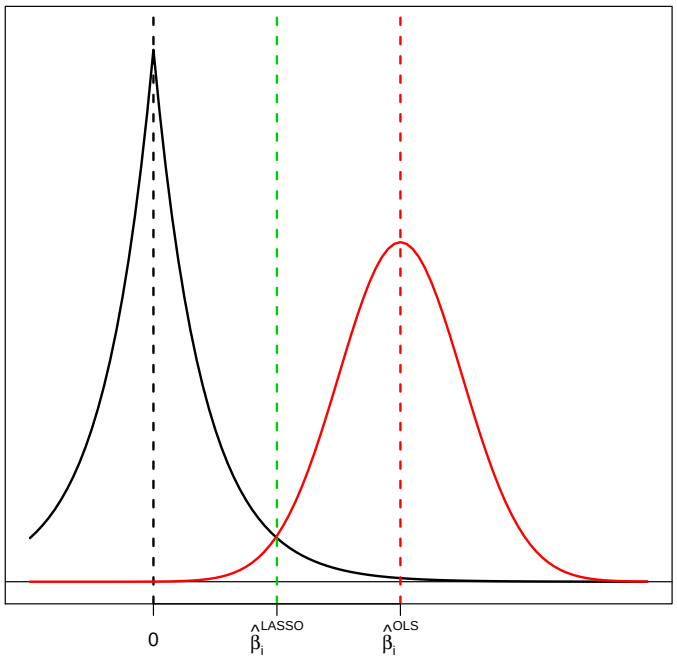
Consider the quadratic error $Error = E[(Y - \hat{f}(X))^2]$ and its decomposition (Hastie et al., 2001)

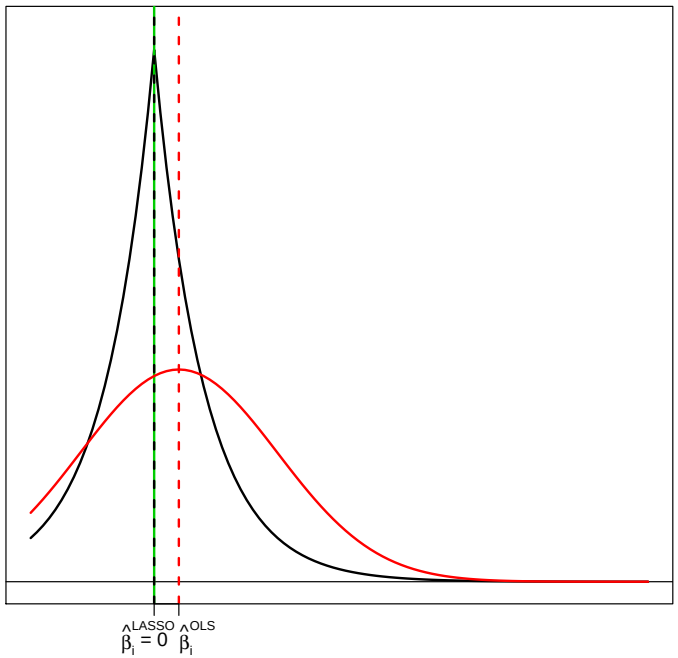
$$E[(Y - \hat{f}(X))^2] = Var(Y) + Bias^2 + Var(\hat{f}(x)) :$$

- an explanatory model aims to minimize the bias (Shmueli, 2010);
- the bias represents the difference between the model and the true mechanism generating the data;
- a prediction model aims to minimize the whole error;
- it may be useful to increase the bias to reduce the variance term:
  - shrinkage;
  - sparsity.
- here we consider two statistical methods with these two properties: lasso (Tibshirani, 1996) and boosting (Friedman, 2001).

Two statistical methods for prediction: lasso I

- penalized regression method:
  $\hat{\beta}^{\text{LASSO}} = \text{argmin}_{\beta} \{||Y - X\beta)||_2 + \lambda||\beta||_1\}$;





- the tuning parameter λ controls the amount of shrinkage and the model sparsity.

Two statistical methods for prediction: lasso II

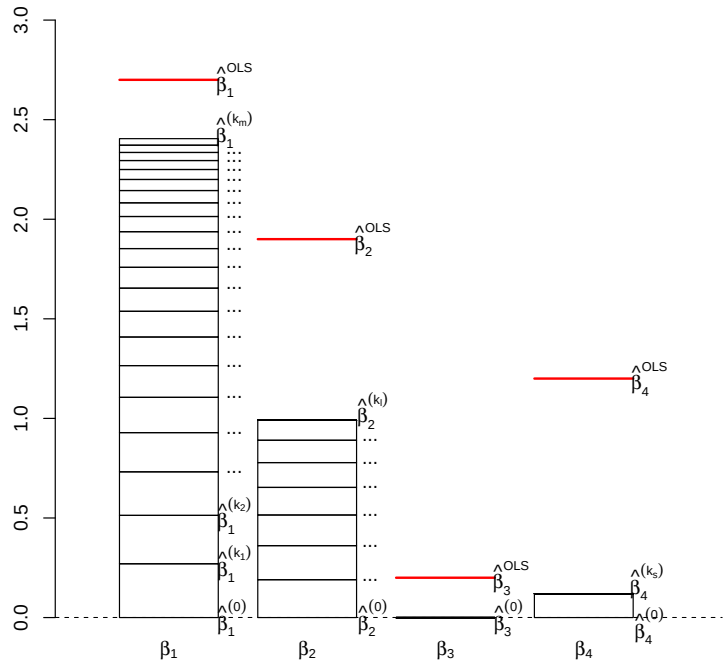
Two well-known R-packages contain functions to implement the lasso:

- *penalized* (Goeman et al., 2014);
- *glmnet* (Friedman et al., 2010).

- here we decided to use the latter:
  - it is a little bit faster;
  - it is more flexible in terms of combining different kinds of data.

Two statistical methods for prediction: component-wise boosting I

- stepwise procedure based on repeated fits of a weak learner to minimize a loss function $L(\cdot)$;



1. $\hat{\beta} = (0, \ldots, 0)$;
2. $u = -\left.\frac{\partial L(y, F(X,\beta))}{\partial F(X,\beta)}\right|_{\beta=\hat{\beta}}$;
3. $\hat{b}_j = \nu \hat{h}(u, X_j)$;
4. $j^* = \min_j L(y, F(X, \hat{b}_j))$;
5. $\hat{\beta}_{j^*} = \hat{\beta}_{j^*} + \hat{b}_{j^*}$.

Repeat 2 – 5 m_{stop} times. The quantity $\nu \hat{h}(\cdot)$ is the weak learner.

- ν is the penalty or "boosting step size";
- m_{stop} denotes the number of boosting iterations and controls the amount of shrinkage and the model sparsity.

Two statistical methods for prediction: component-wise boosting II

Among the R-packages available to perform boosting, we consider:

- *mboost* (Hothorn et al., 2013);
- *GAMBoost* (Binder, 2013b) and its version for time-to-event response variables *CoxBoost* (Binder, 2013a);

These two packages:

- follow two slightly different boosting approaches,
  - model-based (or gradient) boosting (Bühlmann & Yu, 2003);
  - likelihood-based boosting (Tutz & Binder, 2006)
- for similarities and differences, see De Bin (2015).

Two statistical methods for prediction: tuning parameters

- Both lasso and boosting rely on tuning parameters (λ, ν, m_{stop});
- to compute them, we suggest to implement a "repeated cross-validation" procedure:
  - split the sample into K folds (usually 5 or 10);
  - on each fold, evaluate the model trained on the remaining K-1 folds;
  - repeat this procedure several times for different fold splits;
  - average the results;
- the tuning parameter assumes the value associated with the best model performance.

Combining clinical and molecular data in a prediction model: motivation

The inclusion of the clinical information in a prediction model based on high-dimensional molecular (omics) data may have several advantages:

- clinical predictors are often available and usually their predictive value well-validated in the literature;
- clinical data may be cheaper to collect;
- it is possible to focus on the added predictive value of the molecular predictors.

Nevertheless:

- it may be difficult to profitably combine clinical and molecular data;
- issues derived from their different dimensionalities;
- we see some strategies based on the works of Boulesteix & Sauerbrei (2011) and De Bin et al. (2014).

Combining clinical and molecular data in a prediction model: strategies

- **1: "naive"**;
  - treat clinical and molecular data in the same way;
  - issues to fully exploit the clinical information (Binder & Schumacher, 2008; Boulesteix & Sauerbrei, 2011).
- **2: "clinical offset"**;
  - fit a pre-specified clinical model;
  - use the related linear predictor as an offset in fitting a model on the molecular data.
- **3: "favoring"**;
  - in the construction of the prediction model, more importance is somehow given to the clinical part;
  - e.g., in a penalized regression, less/no penalty;

Data: Bauer et al. (2008)'s dataset

- 117 patients with head and neck squamous cell carcinoma;
- response variable: time to local relapse (survival);
- effective sample size (number of events): 49;

- clinical data: *age* ($\leq$ / $>$ 60), *anemia status* (yes/no), *stage* (3 categories), *tumor size* (4), *grade* (2) and *lymph node* (3);
- molecular data: copy number alterations in chromosomes.

- due to missing values, we considered 108 patients (48 events).

Data: split into training and test set

In this dataset we do not have separate training and test sets:

- we need to create them arbitrarily, splitting the data into two sets;
- usual proportions are $\frac{1}{2} - \frac{1}{2}$, $\frac{2}{3} - \frac{1}{3}$ or $\frac{3}{4} - \frac{1}{4}$;
- the analysis should be repeated for several splits;

- ideally, we should have three different sets:
  - one to select the predictors;
  - one to fit the models;
  - one to evaluate the model performances;
- due to scarcity of observations, usually two sets are used.

Models: practical implementation I

- In the following, we use the notation:
  - `clin` = matrix of clinical data;
  - `gene` = matrix of molecular data;
  - `y` = outcome (survival object);
  - the extension "`.t`" denotes quantities of the training set;
  - the extension "`.v`" of the test set;

Strategies implementation:

- **1: "naive"**:
  - lasso:
    ```
    model<-glmnet(x=cbind(clin.t,gene.t),y=y.t,family='cox');
    ```
  - boosting (model-based):
    ```
    model<-glmboost(y=y.t,x=cbind(clin.t,gene.t),family=CoxPH());
    ```
  - boosting (likelihood-based):
    ```
    model<-CoxBoost(time=y.t[,1],status=y.t[,2],
                    x=cbind(clin.t,gene.t),stepno=m_stop);
    ```

Models: practical implementation II

- **2: "clinical offset"**;

```
mod.clin<-coxph(y.t~.,data=as.data.frame(clin.t))
clinical.offset<-mod.clin$linear.predictors
```

  - lasso:

```
model<-glmnet(x=cbind(clin.t,gene.t),y=y.t,
                    family='cox',offset=clinical.offset);
```

  - boosting (model-based):

```
model<-glmboost(y=y.t,x=cbind(clin.t,gene.t),
                    family=CoxPH(),offset=clinical.offset);
```

- **3: "favoring"**;
  - lasso:

```
model<-glmnet(x=cbind(clin.t,gene.t),y=y.t,family='cox',
penalty.factor=c(rep(0,ncol(clin.t)),rep(1,ncol(gene.t))));
```

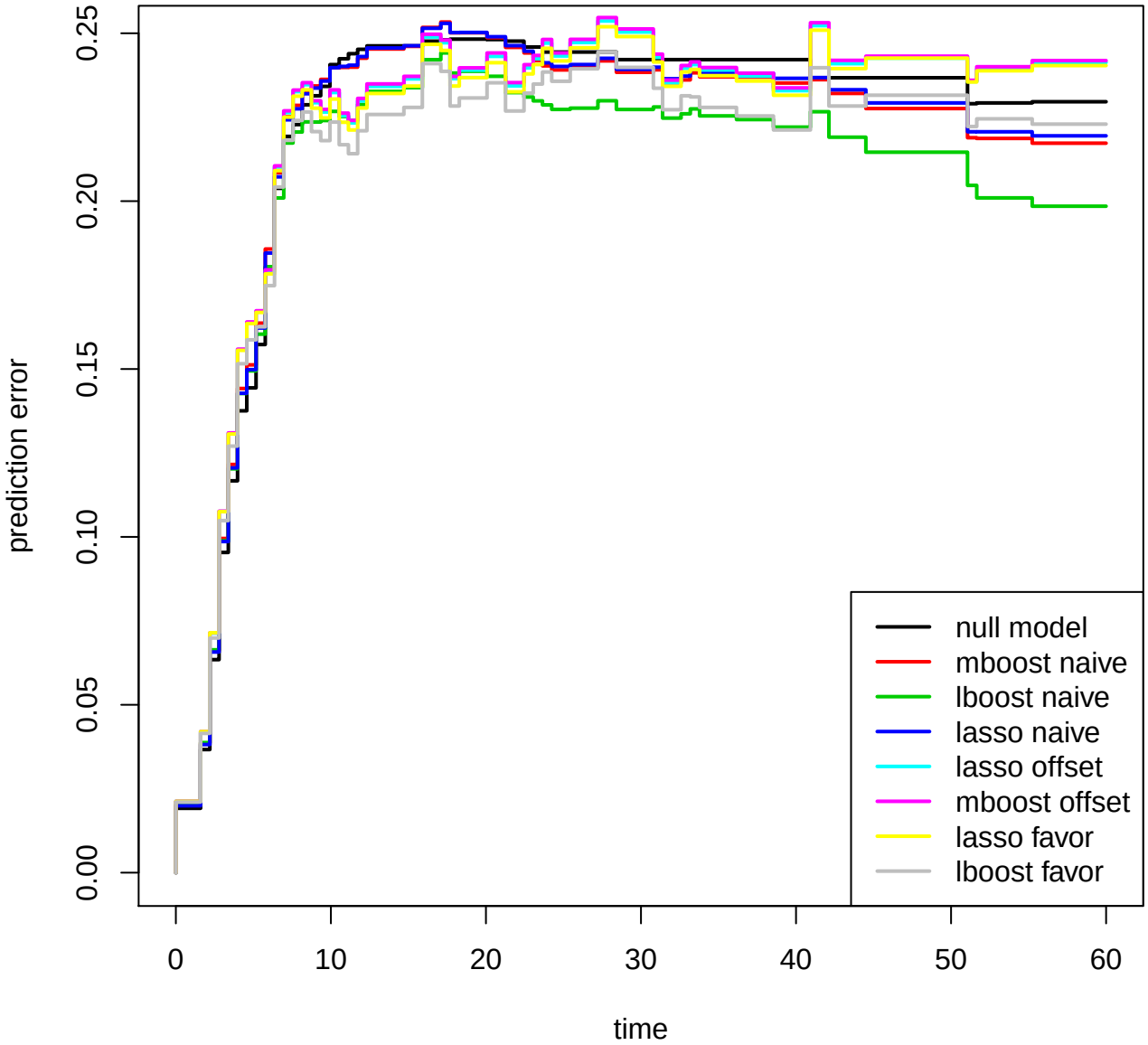
  - boosting (likelihood-based):

```
model<-CoxBoost(time=y.t[,1],status=y.t[,2],x=cbind(clin.t,
            gene.t),stepno=m_stop,unpen.index=1:ncol(clin.t));
```

Model evaluation: test set

- to evaluate the predictive ability of the models, we use the test set;
- training and test set must be completely independent;
- given the time-to-event nature of the data, we compare the models in terms of the Brier score (Graf et al., 1999):
  - measures the goodness of the predicted survival curve;
  - captures both calibration and discrimination ability;
  - idea: if the subject is alive at time t, the predictive survival probability should be close to 1, to 0 otherwise (Schumacher et al., 2007);
  - good predictions correspond to small values of the score.
- we plot the Brier score as a function of the time, using the R-package *pec* (Gerds, 2014).
- we use several (1000) splits into training and test sets.

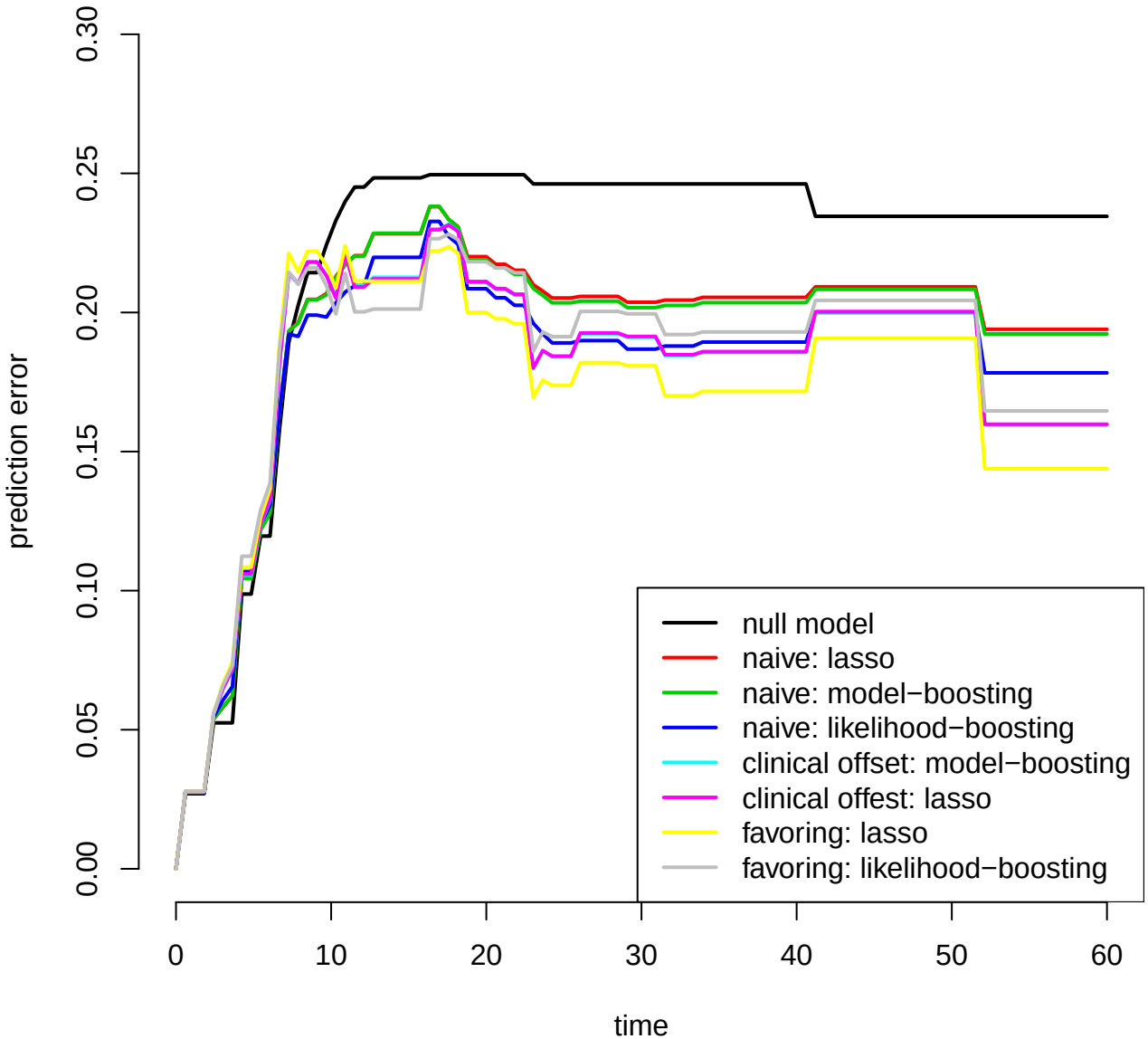
Model evaluation: results



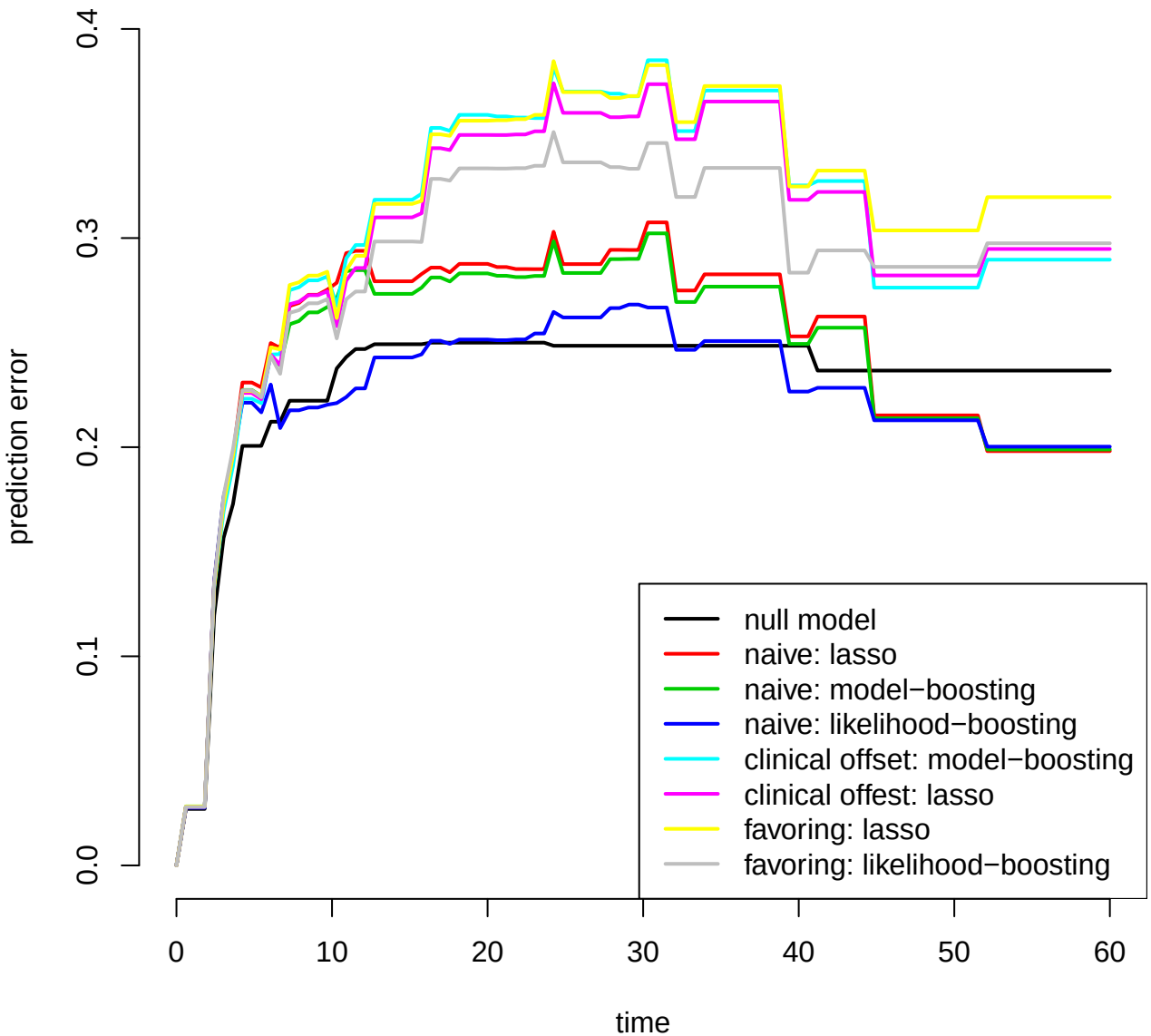
Remarks on model evaluation: the importance of considering several splits

- if we consider only one split into training and test sets, we are limiting the analysis to one specific case:
  - we can be lucky and obtain a great result;
  - we can be unlucky and obtain a bad result;
  - in any case, we are not describing the real situation.
- the smaller the dataset, the more relevant this issue is;
- considering several splits, we limit this issue;
- the results do not depend anymore on a particular split;
- we should average the results of several splits.

Remarks on model evaluation: example of fortunate split



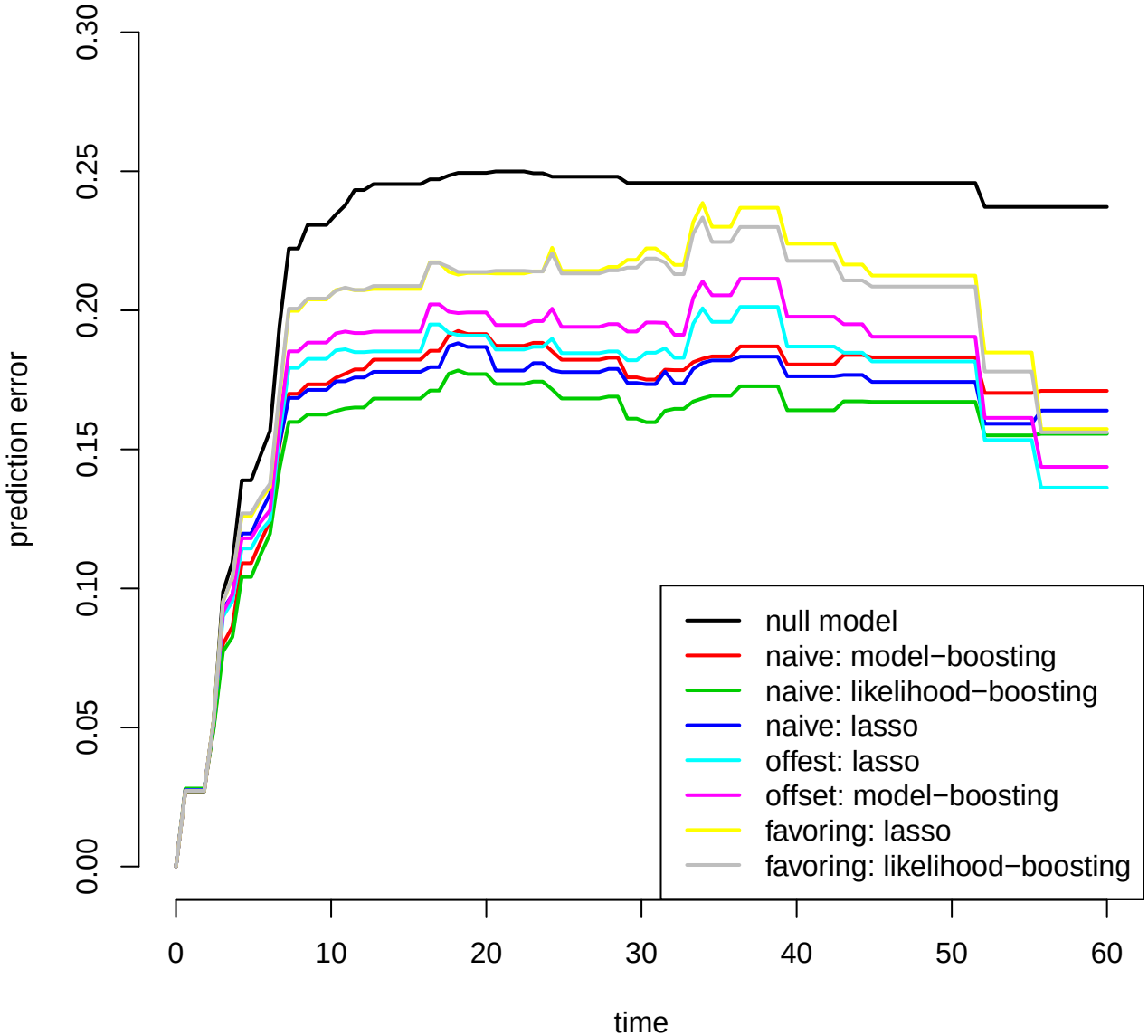
Remarks on model evaluation: example of unfortunate split



Remarks on model evaluation: the independence between training and test sets

- a relevant problem of models based on high-dimensional data is their tendency to overfit the data:
- overfitting occurs when a model explains the random noise instead of the true relationship between predictors and outcome;
- with numerous predictors, minor fluctuations in the data are exaggerated (McShane et al., 2013);
- the computation of the prediction ability of a model requires independent training and test sets:
  - ideally the test set should be a completely different dataset;
  - when a second set is not available, repeated splits.
- the repeated cross-validation procedure to find the values for the tuning parameters must involve only the training set.

Overfitting: example (prediction error computed on the training set)



Conclusions

- we showed some strategies to include clinical information in a prediction model based on high-dimensional data;
- we showed how to implement these strategies using two well-known statistical methods: lasso and boosting;
- we stressed the importance of independent training and test sets;
- in the case of absence of an independent test set, we stressed the necessity of considering several splits of the dataset into training and test sets.

References I

Bauer, V. L., Braselmann, H., Henke, M., Mattern, D., Walch, A., Unger, K., Baudis, M., Lassmann, S., Huber, R., Wienberg, J. et al. (2008). Chromosomal changes characterize head and neck cancer with poor prognosis. *Journal of molecular medicine* **86**, 1353–1365.

Binder, H. (2013a). *CoxBoost: Cox models by likelihood based boosting for a single survival endpoint or competing risks*. R package version 1.4.

Binder, H. (2013b). *GAMBoost: Generalized linear and additive models by likelihood based boosting*. R package version 1.2-3.

Binder, H. & Schumacher, M. (2008). Allowing for mandatory covariates in boosting estimation of sparse high-dimensional survival models. *BMC Bioinformatics* **9**, 14.

Boulesteix, A. L. & Sauerbrei, W. (2011). Added predictive value of high-throughput molecular data to clinical data and its validation. *Briefings in Bioinformatics* **12**, 215–229.

Bühlmann, P. & Yu, B. (2003). Boosting with the L_2 loss: regression and classification. *Journal of the American Statistical Association* **98**, 324–339.

De Bin, R. (2015). Boosting in cox regression: a comparison between the likelihood-based and the model-based approaches with focus on the r-packages coxboost and mboost. Tech. Rep. 180, University of Munich.

De Bin, R., Sauerbrei, W. & Boulesteix, A. L. (2014). Investigating the prediction ability of survival models based on both clinical and omics data: two case studies. *Statistics in Medicine* **33**, 5310–5329.

Friedman, J., Hastie, T. & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* **33**, 1.

References II

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *The Annals of Statistics* **29**, 1189–1232.

Gerds, T. (2014). *pec: Prediction Error Curves for risk prediction models in survival analysis*. R package version 2.4-4.

Goeman, J., Meijer, R. & Chaturvedi, N. (2014). *penalized: L1 (lasso and fused lasso) and L2 (ridge) penalized estimation in GLMs and in the Cox model*.

Graf, E., Schmoor, C., Sauerbrei, W. & Schumacher, M. (1999). Assessment and comparison of prognostic classification schemes for survival data. *Statistics in Medicine* **18**, 2529–2545.

Hastie, T., Tibshirani, R. & Friedman, J. (2001). *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer, New York.

Hothorn, T., Buehlmann, P., Kneib, T., Schmid, M. & Hofner, B. (2013). *mboost: Model-Based Boosting*. R package version 2.4-0.

McShane, L. M., Cavenagh, M. M., Lively, T. G., Eberhard, D. A., Bigbee, W. L., Williams, P. M., Mesirov, J. P., Polley, M.-Y. C., Kim, K. Y., Tricoli, J. V. et al. (2013). Criteria for the use of omics-based predictors in clinical trials: explanation and elaboration. *BMC Medicine* **11**, 220.

Schumacher, M., Binder, H. & Gerds, T. (2007). Assessment of survival prediction models based on microarray data. *Bioinformatics* **23**, 1768–1774.

Shmueli, G. (2010). To explain or to predict? *Statistical Science* , 289–310.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* , 267–288.

Tutz, G. & Binder, H. (2006). Generalized additive modeling with implicit variable selection by likelihood-based boosting. *Biometrics* **62**, 961–971.