Model Validation

Dr. Aurélie Lemmens

Agenda

- Model Validation: Definition and Importance
- In-sample Performance and Overfitting
- The Bias-Variance Trade-off
- The Goal of Modeling: to Explain or to Predict?
- Hold-out Performance: How to Split the Data & Cross-validation
- Choosing Benchmark Models
- Performance Criteria: Loss Function, Significance Testing

Useful readings

- Model validation and the bias-variance trade-off
  - Hastie, Tibshirani, Friedman, Elements of statistical learning, esp. chapter 7: model assessment and selection
  - Shugan (2007) and Shugan (2009) editorials and commentaries
- To explain or to predict
  - Shmueli, G. (2010). To explain or to predict?. *Statistical science*, 289-310.
  - Shmueli, G., & Koppius, O. R. (2011). PREDICTIVE ANALYTICS IN INFORMATION SYSTEMS RESEARCH. MIS Quarterly, 35(3), 553-572.
- Hold-out samples and choice of benchmarks
  - Lemmens, A., Croux, C., & Stremersch, S. (2012). Dynamics in the international market segmentation of new product growth. *International Journal of Research in Marketing*, *29*(1), 81-92.
- Performance criteria
  - Lemmens, A., & Croux, C. (2006). Bagging and boosting classification trees to predict churn. *Journal of Marketing Research*, *43*(2), 276-286.

Model validation:

Definition and importance

Shugan (2007, 2009), editorial and commentary

"Objective evaluation should always trump subjective and opinionated criteria for model evaluation"

- Importance of predictive tests
- "We can easily criticize assumptions as unrealistic when they dismiss our favorite variable, contradict the past literature, or conflict with current beliefs" (Shugan 2007)
- If one would have focused on realism, probably most breakthrough papers would not have been published.

Model validation: definition

- The process of determining the degree to which a [mathematical/computer /…] model is an accurate representation of the real world *from the perspective of the intended model applications*
- Relates to
  - Performance evaluation
  - Model fit
  - Prediction accuracy & prediction errors
- "from the perspective of the intended model applications"

  The application and the goal of the model user should guide us in specifying and evaluating the model

  - *Context* (Shugan 2009): e.g. a model can be relevant for one industry in particular
  - *Goal*: e.g. explanatory purpose vs predictive purpose (see later), e.g. targeting decision

Context drives model evaluation

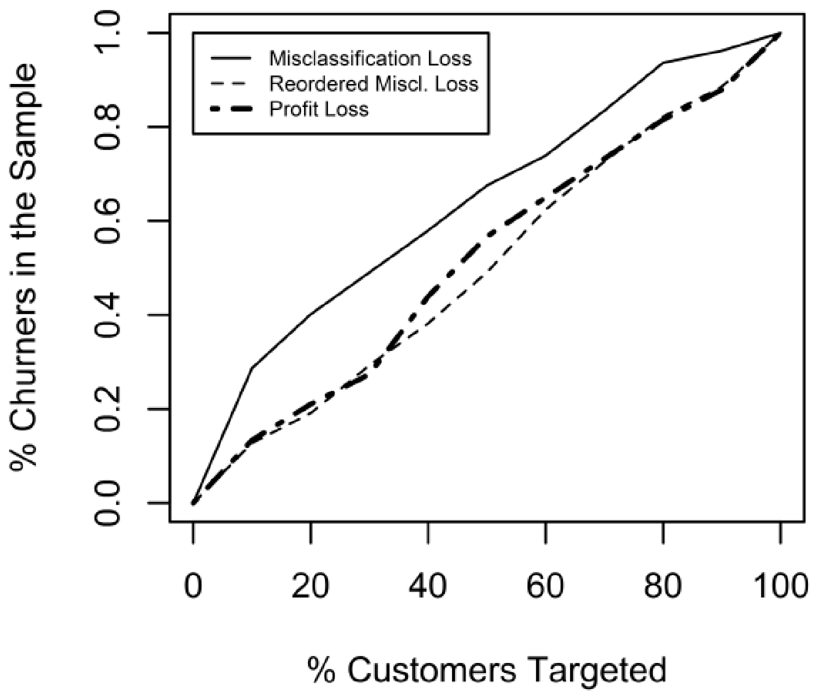
› The following example shows how the particular industry context can influence our assessment on the performance of different prediction models.

	Mean Top-Decile Lift Across 13 industries/applications	Variance in Top-Decile Lift Across 13 industries/applications	Number of Times "Best“ Out of 13 industries/applications
bagging	3.77	6.37	0
bart	3.39	4.08	7
logit	2.94	2.48	2
nb	1.79	0.44	0
nn	2.44	4.12	0
rf	3.75	9.06	3
sgb	3.73	9.04	2
svm	2.13	5.71	0
tree	3.49	5.64	2

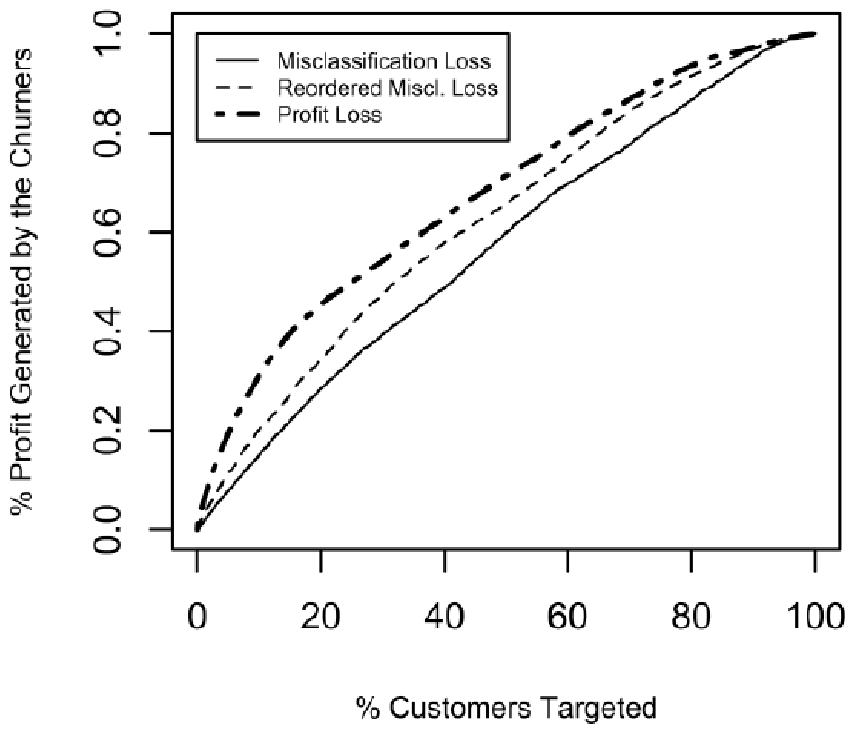
Goal drives model evaluation

- The example shows that one model can be best at rank-ordering customers and at the same time be worst at targeting high profit customers

ROC curve



Profit curve



Importance of validation

- For practitioners
  - Need to choose the best model (which model should I select?)
  - Measure accuracy/power of selected model (can I base my decisions on this model?)
  - Good to measure ROI of the modeling project (how much income do these decisions generate for the company?)
- For academics
  - Estimation methods are inherently designed to minimize a specific “loss”. Evaluating a model allows to test whether this goal is achieved.
    - Alignment estimation and evaluation criterion!
  - To an extent, a model will always fit “noise” as well as “signal”. The goal of evaluation is to investigate to what extent the model captures this “signal”.

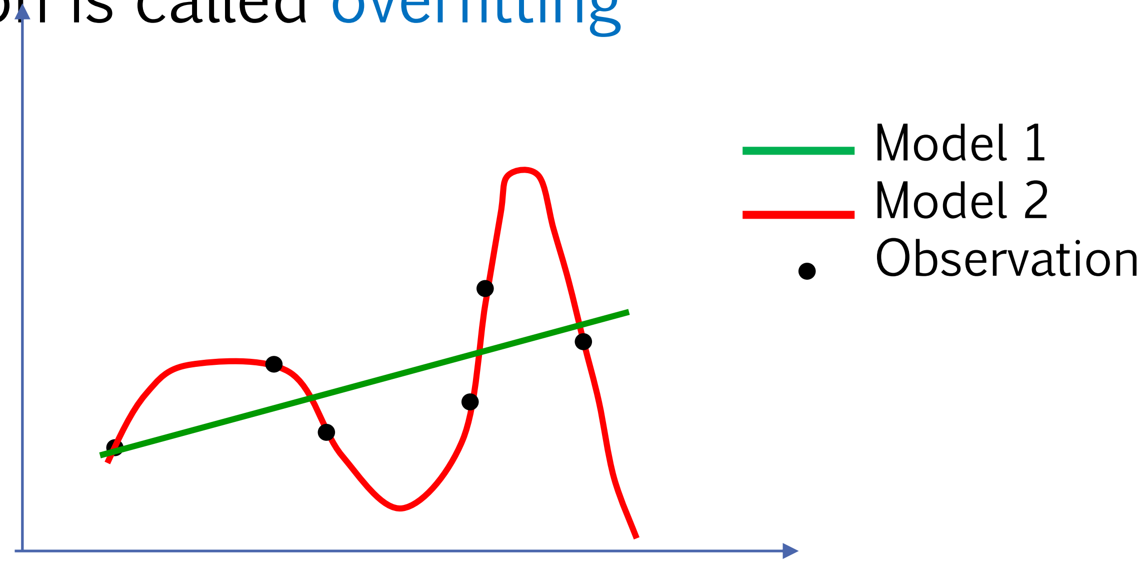
➔ If you fit a number of models on a given dataset and choose the “best” one, it will likely be overly “optimistic”.

In-sample performance:

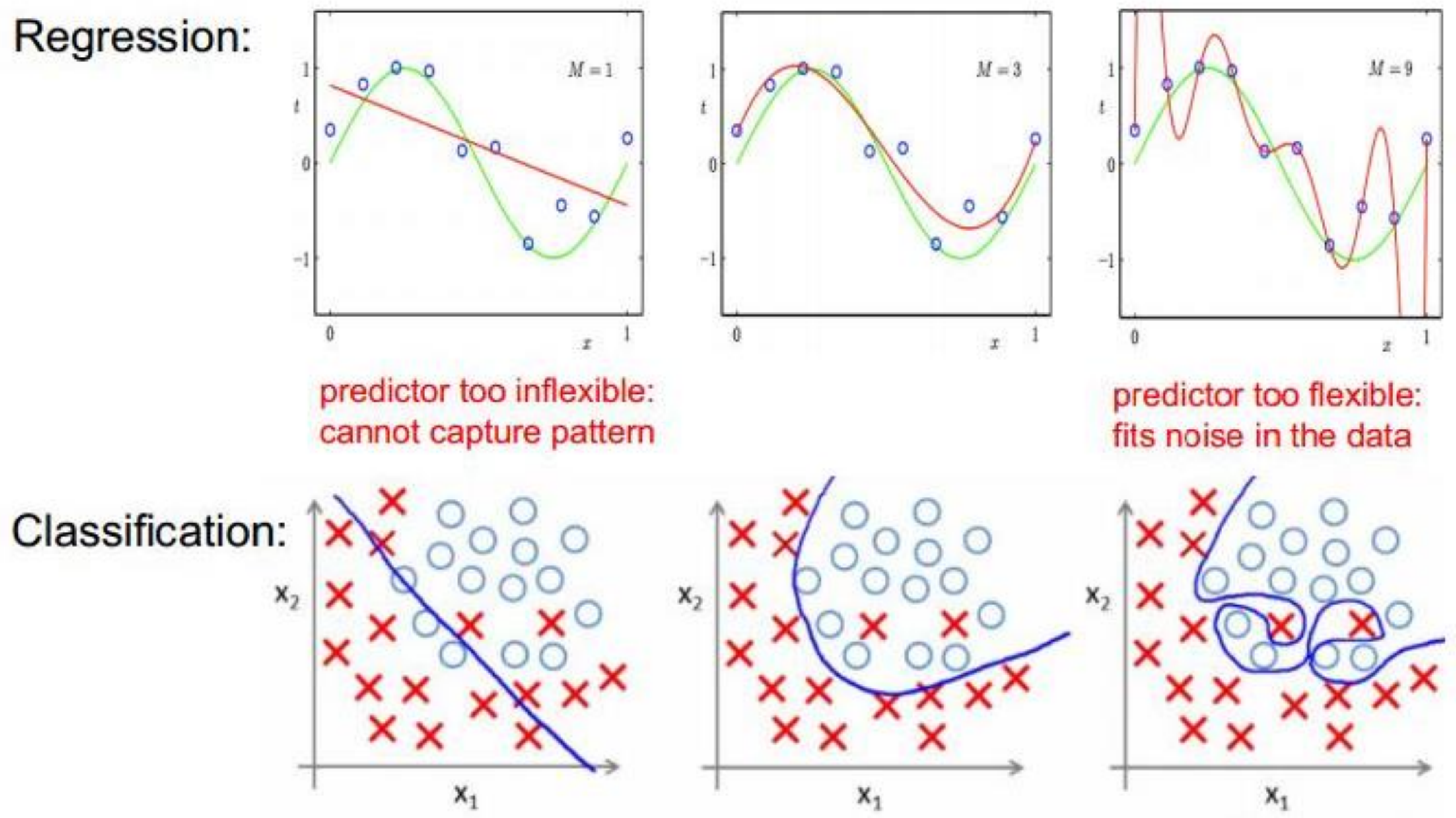
The problem of overfitting

In-sample performance

- Estimate a model on a given sample & evaluate its *in-sample* performance on the same sample
  - By opposition to *holdout* or *out-of-sample performance*
- When in-sample fit criteria are used for model evaluation and model selection, one is likely to obtain a model that consider the "idiosyncrasies/noise" in the data as "useful" information
- This phenomenon is called overfitting



Under- and overfitting examples



The problem of overfitting

- Fitting noise/idiosyncrasies as signal
- This is problematic as a model should achieve a good generalization to new cases
  - Good in-sample performance does not mean good out-of-sample performance
- When does overfitting occur?

1. Focus on in-sample performance
2. Increased model complexity (flexibility) *relative to the sample size*
3. High parameter instability, often resulting from (2)
4. Model selection among many candidates (multiple testing problem)

Model complexity in pursuit of realism?

- Shugan (2007) explains that

> *"Making supposedly more realistic assumptions often results in more variables, relationships, indeterminacy, and complexity."*

- Shugan (2009)

> *"Adding complexity only for the sake of realism defeats the objective of modeling, risks overfitting, and nullifies the benefits from abstraction."*

Bias-variance trade-off

The role of model complexity

Bias & variance: conceptual definitions

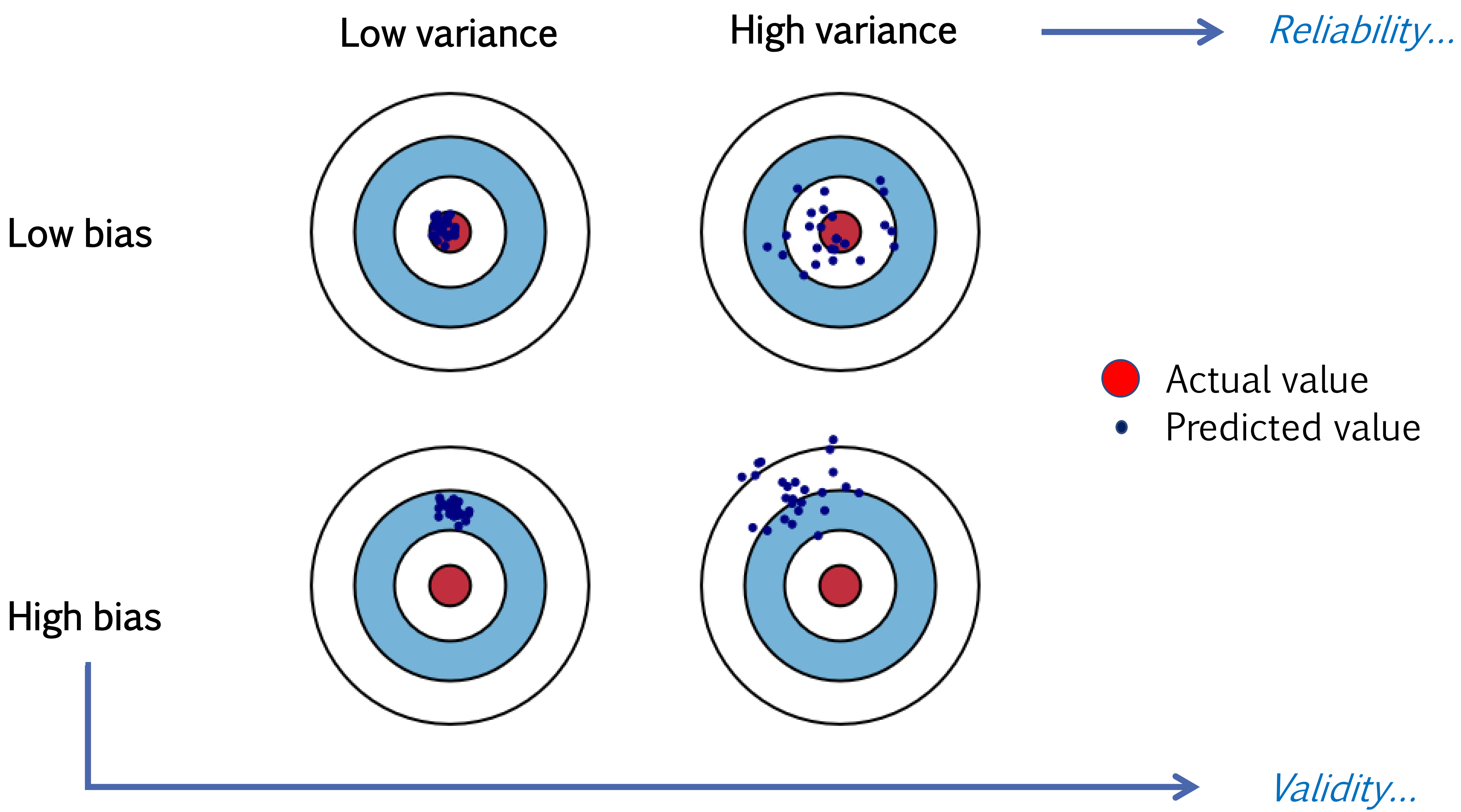
- **Model bias:** $E\left(Y - \hat{f}(x_0)\right)$
  - Difference between the *expected (or average)* prediction of the model and the actual value of the data:
  - Driven by model mis-specification

➔ The model bias measures how far off a model predictions are from the correct value

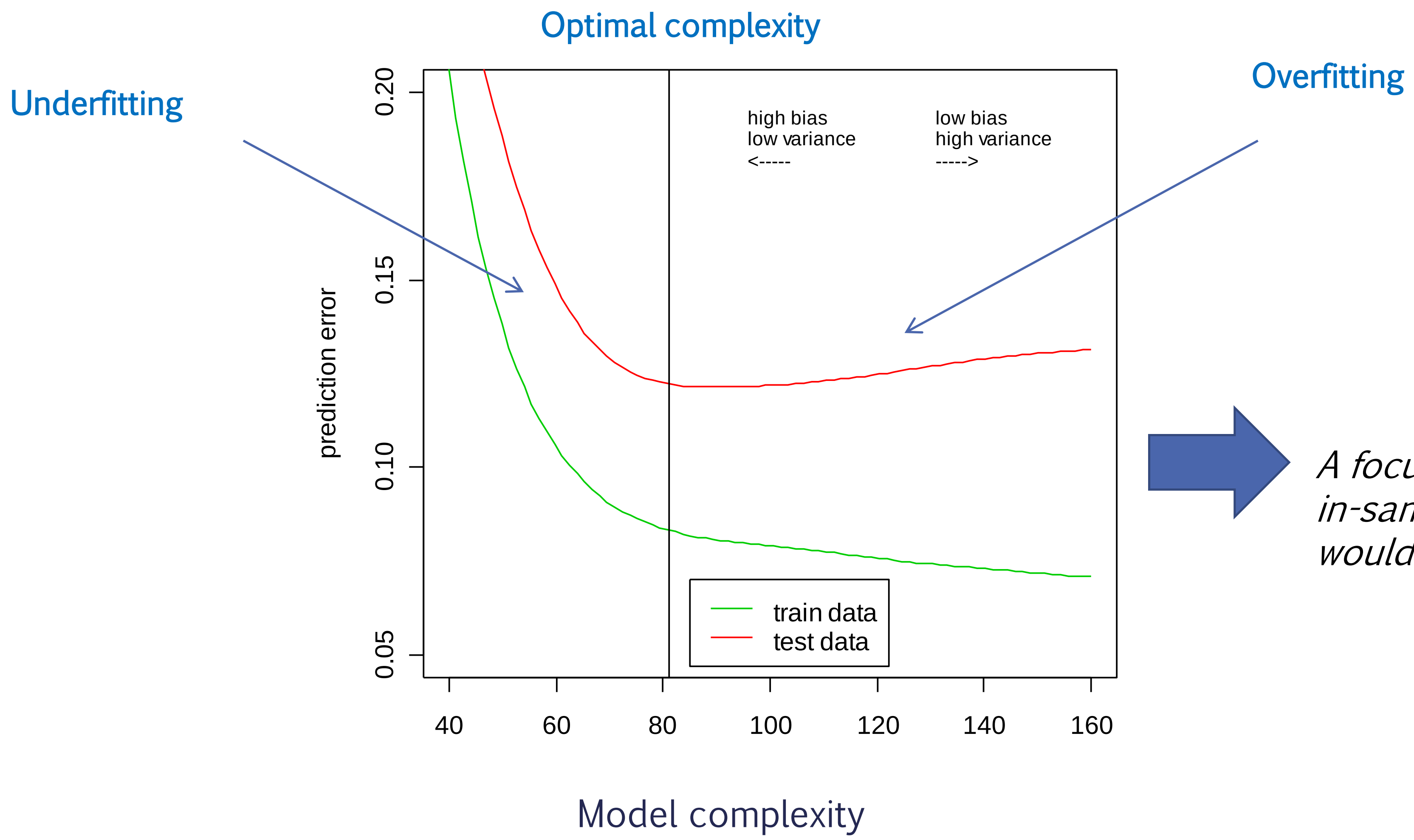
- **Sampling variance:** $\mathrm{var}(\hat{f}(x_0))$
  - Variability of a model prediction for a given observation
  - Driven by the estimation sample (size) and randomization

➔ The sampling variance is how much the predictions for a given point vary between different realizations of the model

Bias-variance: visual representation



Model complexity: bias-variance tradeoff

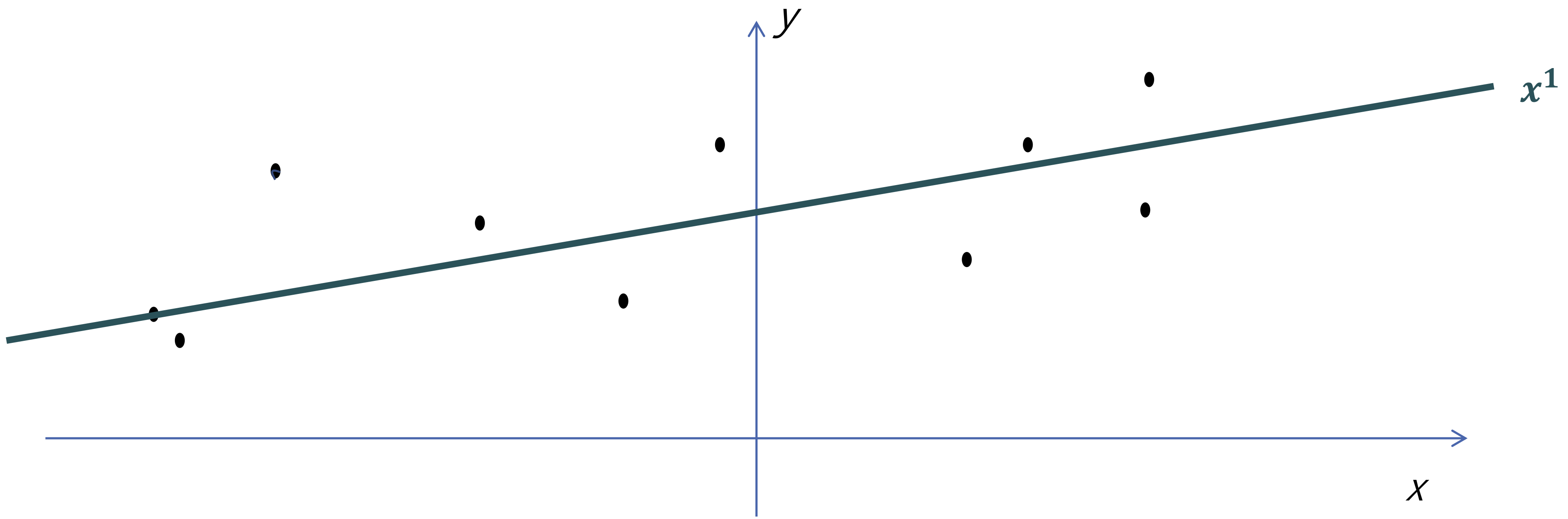


A focus on the in-sample performance would result in overfitting

Hastie, Tibshirani, Friedman "Elements of Statistical Learning" 2001

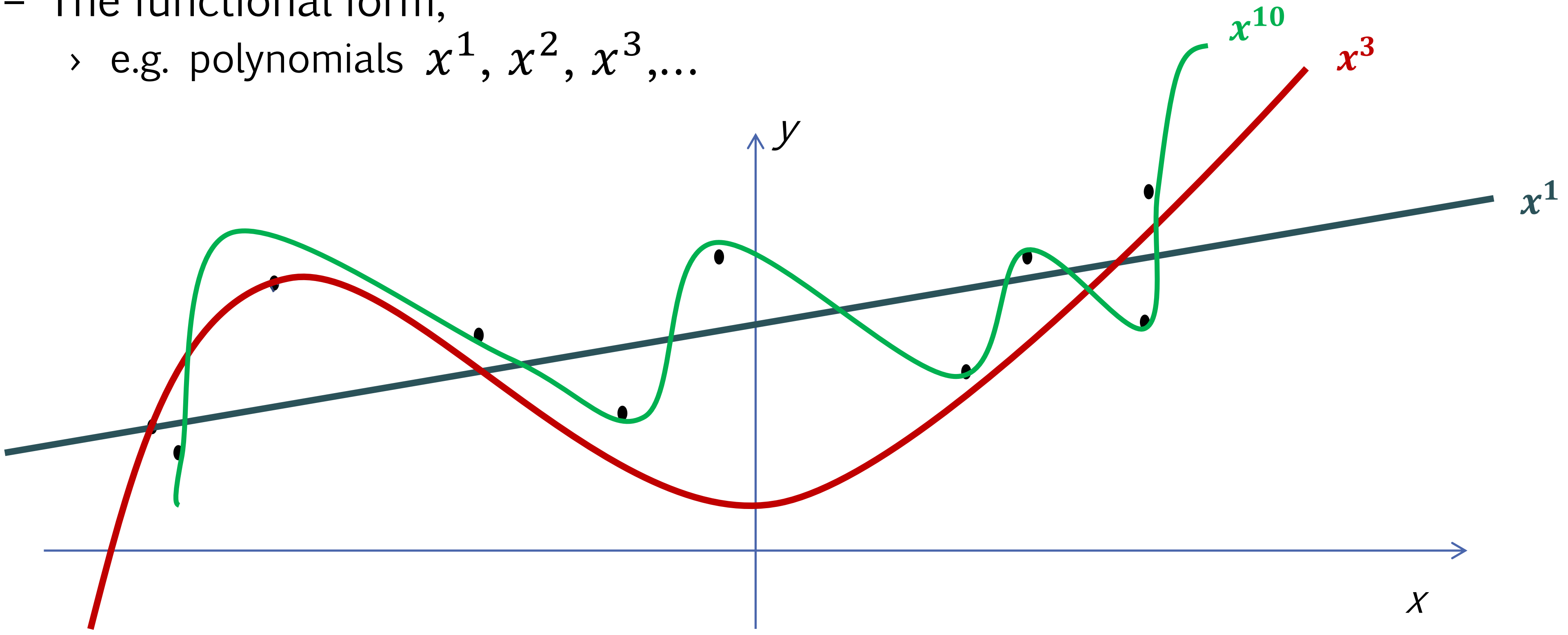
Complexity of a model

- The complexity of a model is a function of
  - The number of parameters (e.g. number of explanatory variables)
  - The number of segments (e.g. latent-class)
  - The functional form,
    - e.g. polynomials $x^1, x^2, x^3, \ldots$



Complexity of a model

- The complexity of a model is a function of
  - The number of parameters (e.g. number of explanatory variables)
  - The number of segments (e.g. latent-class)
  - The functional form,
    - e.g. polynomials x^1, x^2, x^3,…



R code: classification example

```
library(MASS)
set.seed(545)
n=1000 #number of observations
p=10  #number of independent variables
Sigma=matrix(0.3,p,p)   #correlation matrix between the independent variables
diag(Sigma)=rep(1,p)
beta = rnorm(p+1,0,1)

#generate the x and y variables
#===============================
x = cbind(1,mvrnorm(n,rep(0,p),Sigma))
prob = exp(x%*%beta )/(1 + exp(x%*%beta ))  #probability of success for each observation
y = rbinom(n=n, size=1, prob=prob)      #draw whether obs. is a success given a prob.
my.data = data.frame(x=x,y=y)

#estimate a logit model
#===============================
mylogit=glm(factor(y)~-1+.,as.data.frame(my.data),family=binomial(link="logit"))
pred=predict(mylogit,type="response",as.data.frame(my.data))

topf(y=my.data$y,p=pred,share=.1)
ginif(y=my.data$y,p=pred)
```

Bias-variance tradeoff

- Prediction errors can be decomposed into two components
  - Error due to *model bias*
  - Error due to *sampling variance*
- There is a tradeoff between a model's ability to minimize bias and variance (Friedman 1997)
- Understanding these two types of errors helps avoiding under-fitting and overfitting
- *Complexity decreases the bias but increases the variance*

Bias-variance: mathematical definition

- Assume $Y = f(X) + \varepsilon, \quad \varepsilon \sim N(0, \sigma_\varepsilon{}^2)$
- We estimate a model $\hat{f}(X)$
- The expected squared prediction error is $MSE\left(\hat{f}(x_0)\right) = E\left[(Y - \hat{f}(x_0))^2\right]$
- This error can be decomposed into bias and variance:

$$MSE\left(\hat{f}(x_0)\right) = \sigma_\varepsilon{}^2 + Bias^2(\hat{f}(x_0)) + Var(\hat{f}(x_0))$$

Noise
Lower bound on performance

(Expected error due to model mismatch)²

Variation due to the estimation sample and randomization

Proof

$$\text{Remember}: \mathrm{var}(A) = E\left(A^2\right) - \left[E(A)\right]^2$$

$$\text{If we set}: A = Y - \hat{f}(x_0), \quad \text{we obtain}:$$

$$\mathrm{var}(Y - \hat{f}(x_0)) = E\left[\left(Y - \hat{f}(x_0)\right)^2\right] - \left[E\left(Y - \hat{f}(x_0)\right)\right]^2$$

where each term can be rewritten as:

$$\mathrm{var}(Y - \hat{f}(x_0)) = \mathrm{var}(f(x_0) + \varepsilon - \hat{f}(x_0)) = \sigma_\varepsilon^2 + \mathrm{var}(\hat{f}(x_0))$$

$$E\left[\left(Y - \hat{f}(x_0)\right)^2\right] = \mathrm{MSE}\left(\hat{f}(x_0)\right)$$

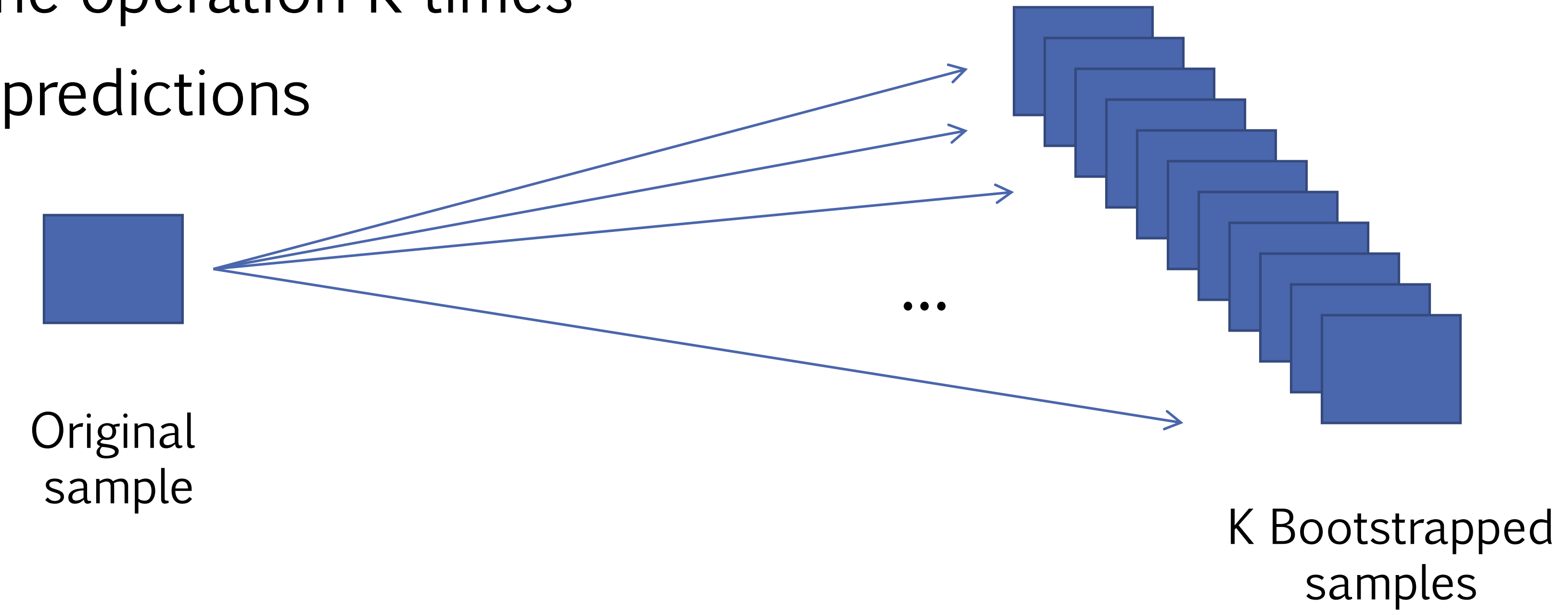
$$\begin{aligned}\left[E\left(Y - \hat{f}(x_0)\right)\right]^2 &= \left[E(Y) - E\left(\hat{f}(x_0)\right)\right]^2 \\ &= \left[f(x_0) - E\left(\hat{f}(x_0)\right)\right]^2 \\ &= \left[-\mathrm{bias}\left(\hat{f}(x_0)\right)\right]^2 \\ &= \mathrm{bias}\left(\hat{f}(x_0)\right)^2\end{aligned}$$

Stylized example

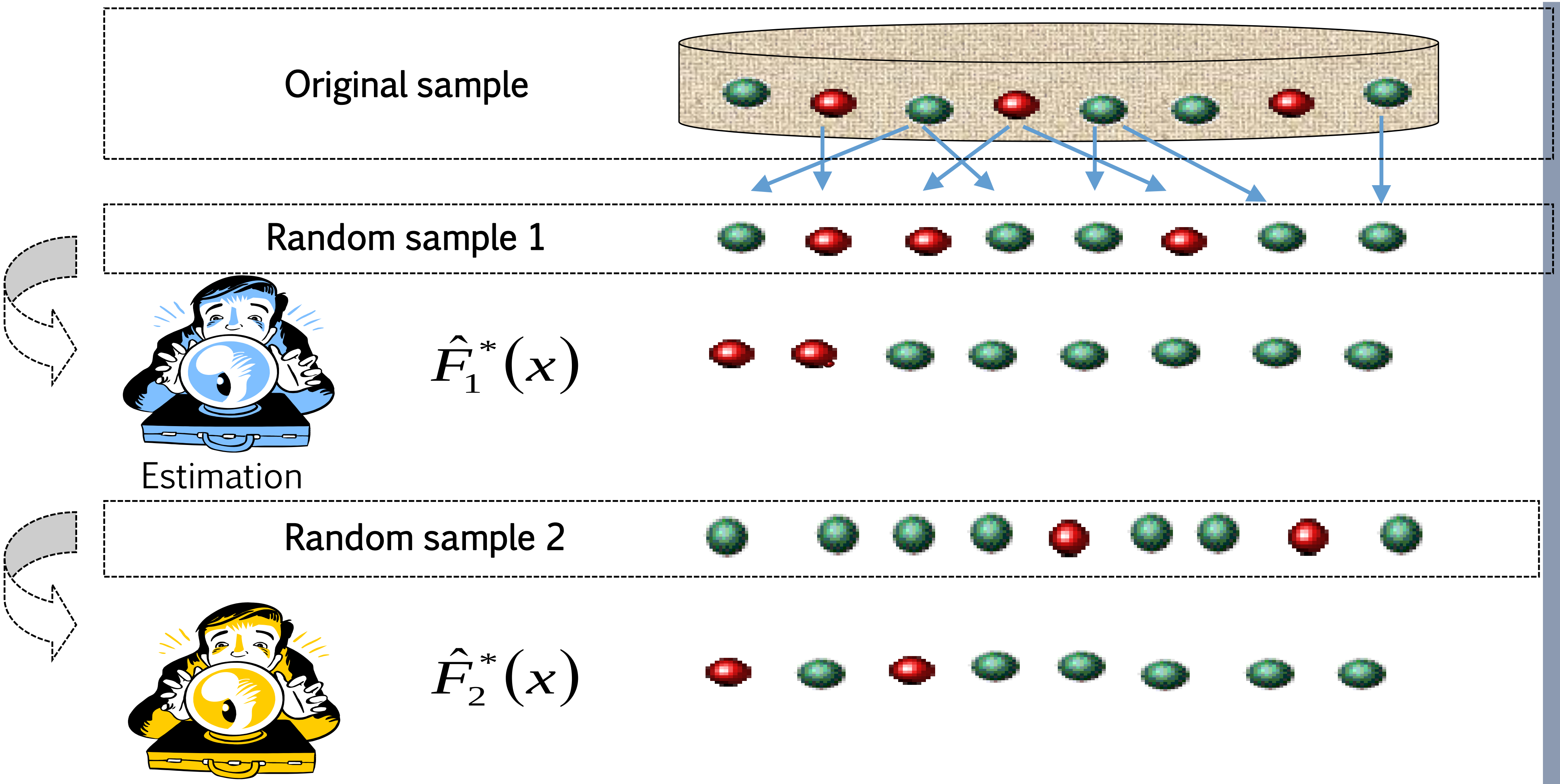
- <u>**Question**</u>: what is the customer defection probability at T-Mobile?
- <u>**Procedure**</u>: survey among 50 T-Mobile customers asking respondents whether they intend to defect.
  - 13 say yes
  - 16 say no
  - 21 do not respond
- <u>**Estimation**</u>: defection probability = 13/(13+16) = 44.8%
- <u>**What is wrong**</u>?
  - Source of error due to bias: (i) non-response, (ii) intention vs. actual behavior, (iii) other response style biases
  - Source of error due to variance: small sample size

Use Bootstrapping to compute bias and variance

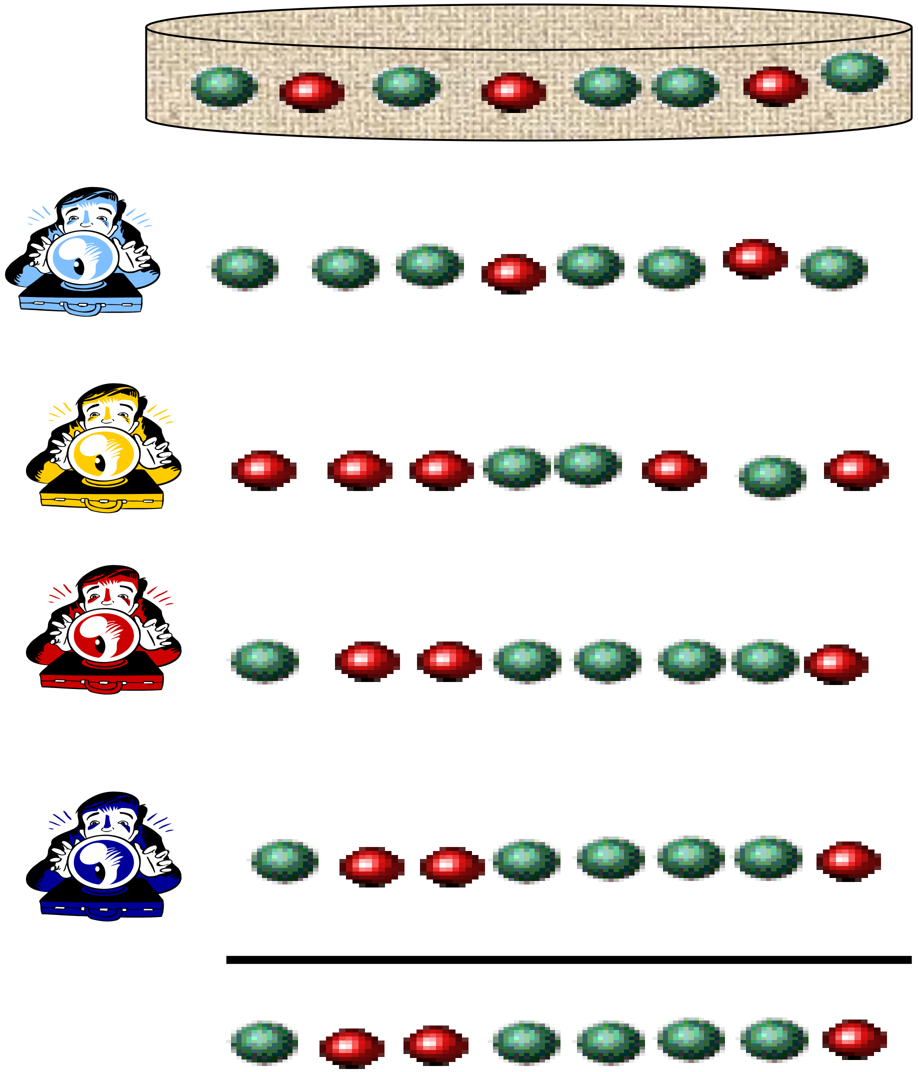
- Draw with replacement N observations from the original sample of size N
- Estimate the model on the bootstrapped sample
  - Sample size of one estimation sample = N
- Repeat the operation K times
- Average predictions



Bootstrapping



Combining predictions



$$\hat{F}_1^*(x)$$

$$\hat{F}_2^*(x)$$

$$\hat{F}_3^*(x)$$

$$\dots$$

$$\hat{F}_K^*(x)$$

$$\hat{F}_b(x) = \frac{1}{K}\sum_{k=1}^{K}\hat{F}_k^*(x)$$

Use Bootstrapping to compute bias and variance

```
S=100
train.index=1:floor(n/2)
test.index=(floor(n/2)+1):n
my.data.train=my.data[train.index,]
my.data.test=my.data[test.index,]
pred.test=array(NA,c(nrow(my.data.test),S))

for (s in 1:S)
{
  bootstrap.index=sample(1:nrow(my.data.train),nrow(my.data.train),replace=TRUE)
  my.boot.data=my.data.train[bootstrap.index,]
  mylogit=glm(factor(y)~-1+.,as.data.frame(my.boot.data),family=binomial(link="logit"))
  pred.test[,s]=predict(mylogit,type="response",as.data.frame(my.data.test))
}
average.pred.test=apply(pred.test,1,mean)
est.biassquared=mean((average.pred.test-my.data.test$y)^2)
est.variance=mean(apply(pred.test,1,var))
est.MSE=mean(apply((pred.test-my.data.test$y)^2,1,mean))
est.MSE
est.variance+est.biassquared
```

Exercise

- Create bootstrapped standard errors for the parameter estimates and compare with the ones obtained from the GLM function

The goal of modeling:

To explain or to predict?

Choosing for bias versus variance

- "Gut feeling" often pushes researchers to minimize bias even at the expense of variance

"Something is wrong with the model"

- They acknowledge that sampling variance is also problematic but

"A model with high variance is not fundamentally wrong if it predicts well on average"

- Be careful: in practice, one deals with *one* realization of the model (one dataset)

To explain or to predict? Shmueli and Koppius (2011)

"The goal of finding a predictively accurate model differs from the goal of finding the true model"

- Explanatory modeling
  - Testing **causal** theory
  - The model needs to be good at recovering the "true" relationships between variables.
  - Goal: minimizing bias ➔ Favor in-sample fit measures and good statistical properties of the estimator
- Predictive modeling
  - Applying a model to predict/forecast new or future observations
  - The model needs to be good at predicting the future, other data sets,…
  - Goal: minimizing the combination of bias and variance ➔ Favor generalization, i.e. out-of-sample performance
- Example: endogeneity and models that "correct for it" (Ebbes et al. 2011)

To explain or to predict?

- Shmueli (2010)

"… predictive modeling is often valued for its applied utility, yet is discarded for scientific purposes such as theory building or testing. Shmueli and Koppius (2010) illustrated the lack of predictive modeling in the field of IS. Searching the 1072 papers published in the two top-rated journals *Information Systems Research* and *MIS Quarterly* between 1990 and 2006, they found only 52 empirical papers with predictive claims, of which only seven carried out proper predictive modeling or testing…"

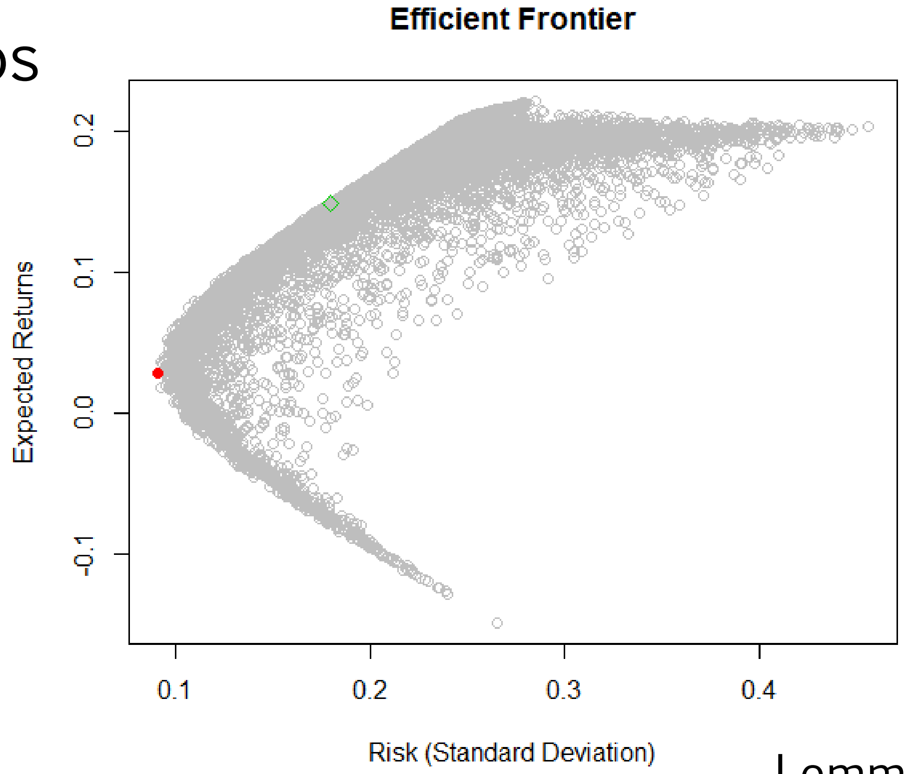
Golden rules to remember:

1. Assess the stability of the parameters' value
2. Validate models on a holdout sample
3. Use performance criteria who penalize complexity (e.g. adjusted R2, information criteria such as BIC, CAIC, ...). Favor simpler models with fewer parameters
4. Eliminate outliers or use methods robust to outliers (robust estimators)
5. Introduce stochasticity and average across many realizations of the model (e.g. bootstrapping, cross-validation, bagging, random forests)
6. Train the model less long and/or add a learning rate parameter (e.g. SGB)

➔ **find the sweet spot,** i.e. the level of complexity at which the increase in bias is equivalent to the reduction in variance.

Example: Model Averaging

- Combining predictions from various models reduces the variance in predictive performance while ensuring a good expected predictive performance overall.
- Combine the benefits of various models into one.
- Analogy to financial portfolios



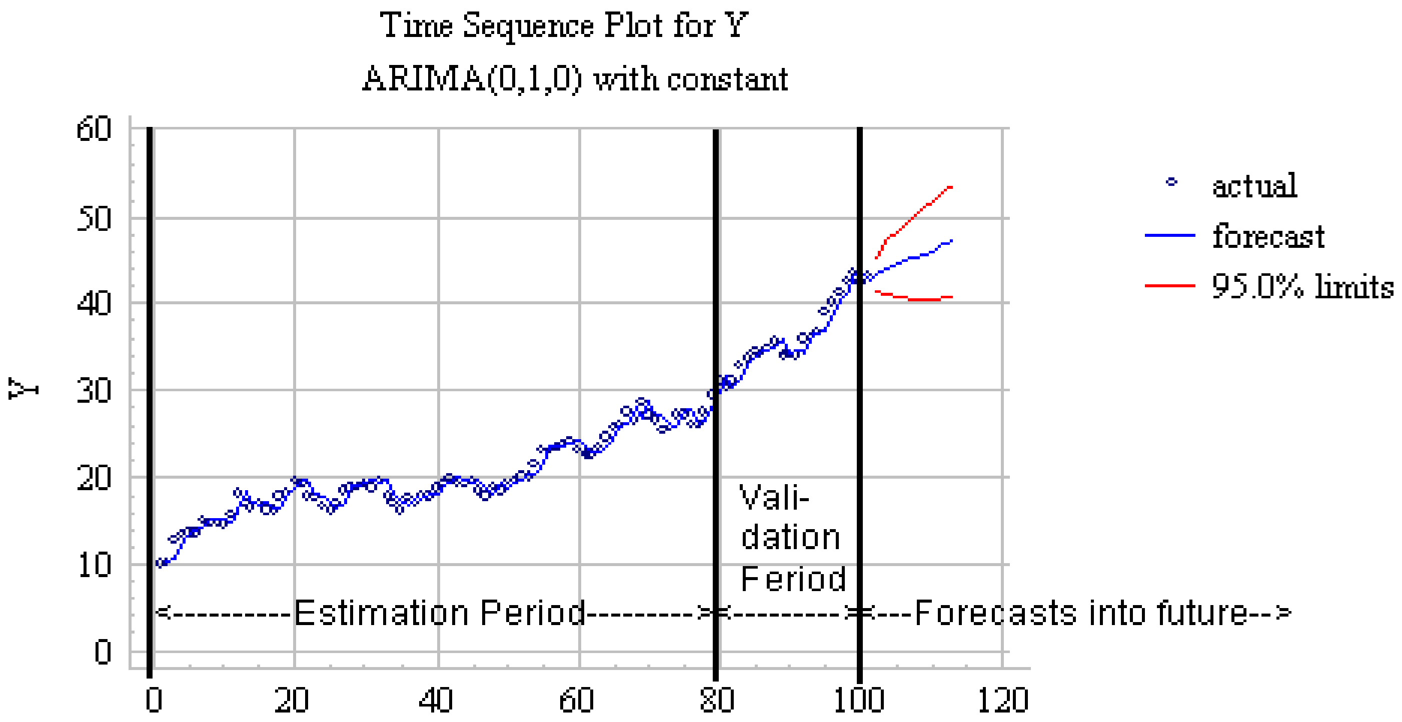
Lemmens et al. (2017)

Holdout performance:

How to split the data and cross-validation

Holdout sample and out-of-sample validation

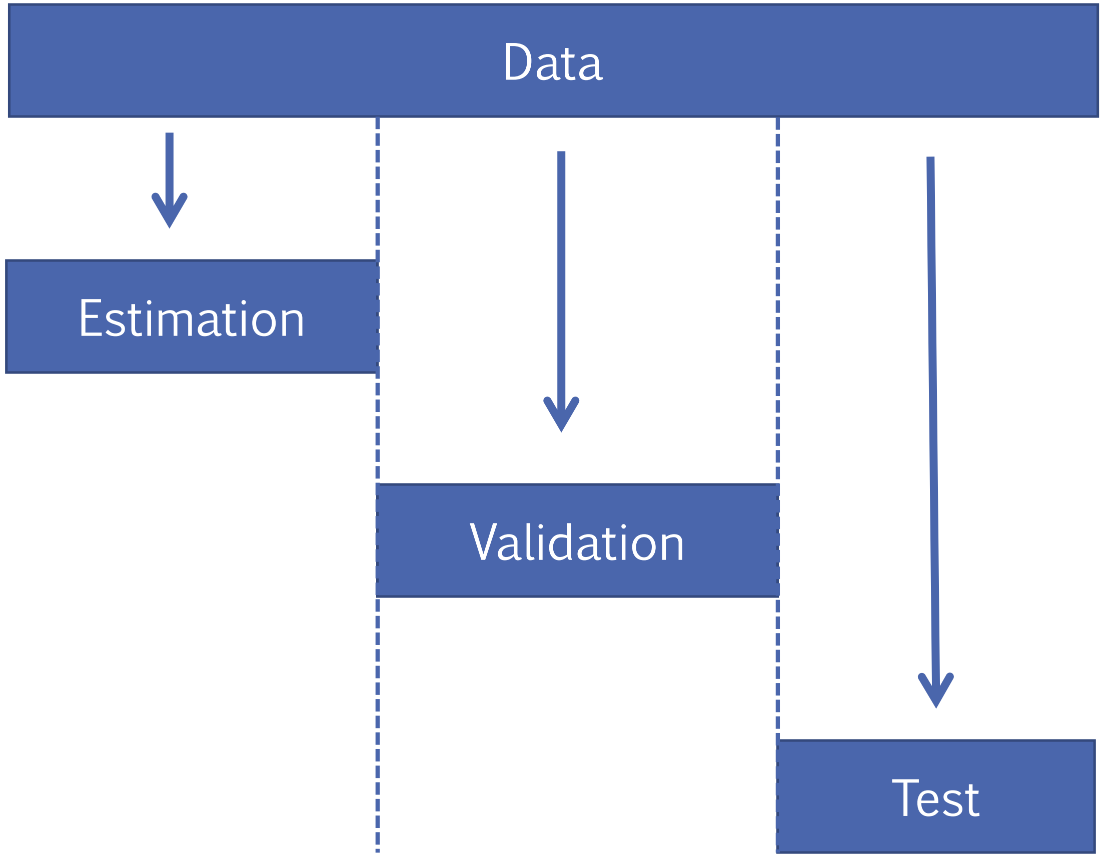
- Withhold some of the sample data from the model identification and estimation process, then use the model to make predictions for the hold-out data in order to see how accurate they are and to determine whether the statistics of their errors are similar to those that the model made within the sample of data that was fitted.



http://people.duke.edu/~rnau/three.htm

Estimation, validation and testing

- Estimation sample
- Validation sample
- Test sample



Estimation sample

- Also called calibration sample or training sample
- Used to estimate the model parameters (and for model selection)
- In-sample forecasts are called fitted values
- Performance is referred to as model fit or in-sample performance
- Forecasts are not completely “honest” as the data on both sides are used twice ➔ overfitting

- True prediction error = in-sample error + optimism error

Validation sample

- Held out during estimation
- In theory, it should be a hold-out sample and forecasts would then be “honest” but it depends…

  ➔ *contamination risk*

- …(Often) used for model validation and selection
- *Test for overfitting:* If the data have not been overfitted, performance on the validation sample should be similar or slightly higher than in-sample performance

Test sample

- Kept away during model estimation and selection

  ➔ *no contamination*

- Used for model performance testing and comparison

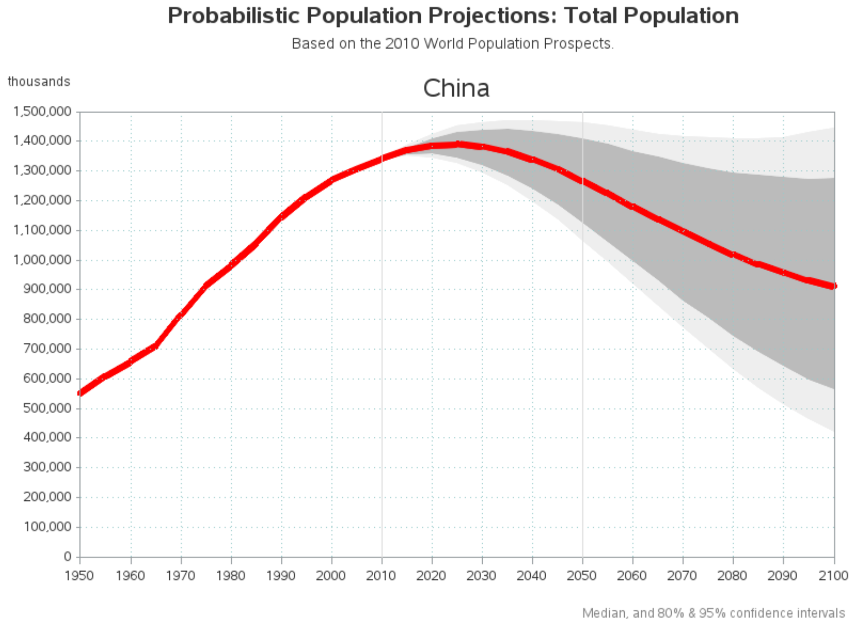
- Forecasts into the future are "true" forecasts

- Check Netflix tournaments, or KDD Cups (https://www.cs.uic.edu/~liub/Netflix-KDD-Cup-2007.html)


Netflix Prize
COMPLETED
Home
Rules
Leaderboard
Update
Congratulations!
The Netflix Prize sought to substantially improve the accuracy of predictions about how much someone is going to enjoy a movie based on their movie preferences.
On September 21, 2009 we awarded the $1M Grand Prize to team "BellKor's Pragmatic Chaos". Read about their algorithm, checkout team scores on the Leaderboard, and join the discussions on the Forum.
We applaud all the contributors to this quest, which improves our ability to connect people to the movies they love.

Test sample

- How to deal with forecasts of an extrapolative model?
  - $Y_t = f(Y_{t-1},\dots, Y_{t-k})$:
  - Succession of one-step ahead predictions
  - $Y_t = f(X_{t-1},\dots, X_{t-k})$:
  - first predict future values of X and then predict future values of Y
  - Confidence intervals typically widen as the forecast horizon expands due to the build-up of error at every time period.

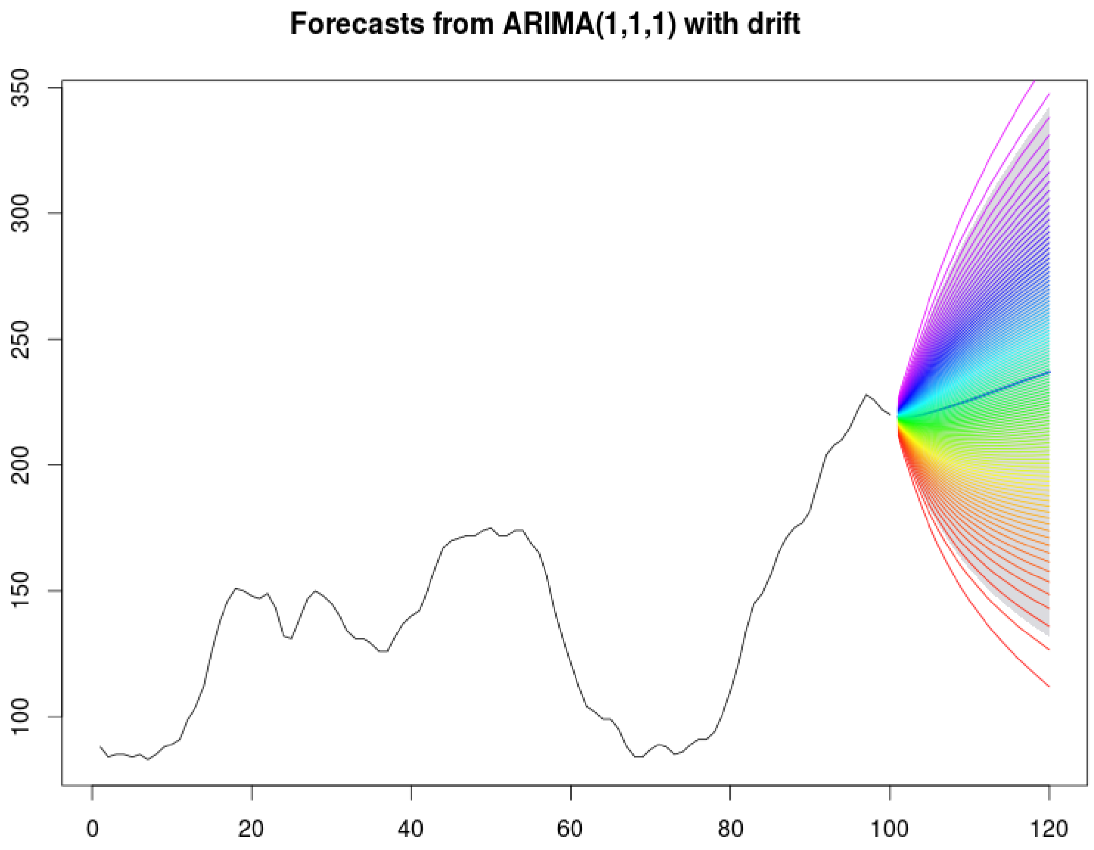


Prediction intervals

- Assume a Normal distribution

```
library(forecast)
fit <- auto.arima(WWWusage)
fc <- forecast(fit, h=20, level=95)     #20 period ahead forecasts
qf <- matrix(0, nrow=99, ncol=20)
m <- fc$mean                            # mean
s <- (fc$upper-fc$lower)/1.96/2         # standard deviation
for(h in 1:20)
qf[,h] <- qnorm((1:99)/100, m[h], s[h]) #generate quantile for every probability level
plot(fc)
matlines(101:120, t(qf), col=rainbow(99), lty=1)
```

Prediction intervals

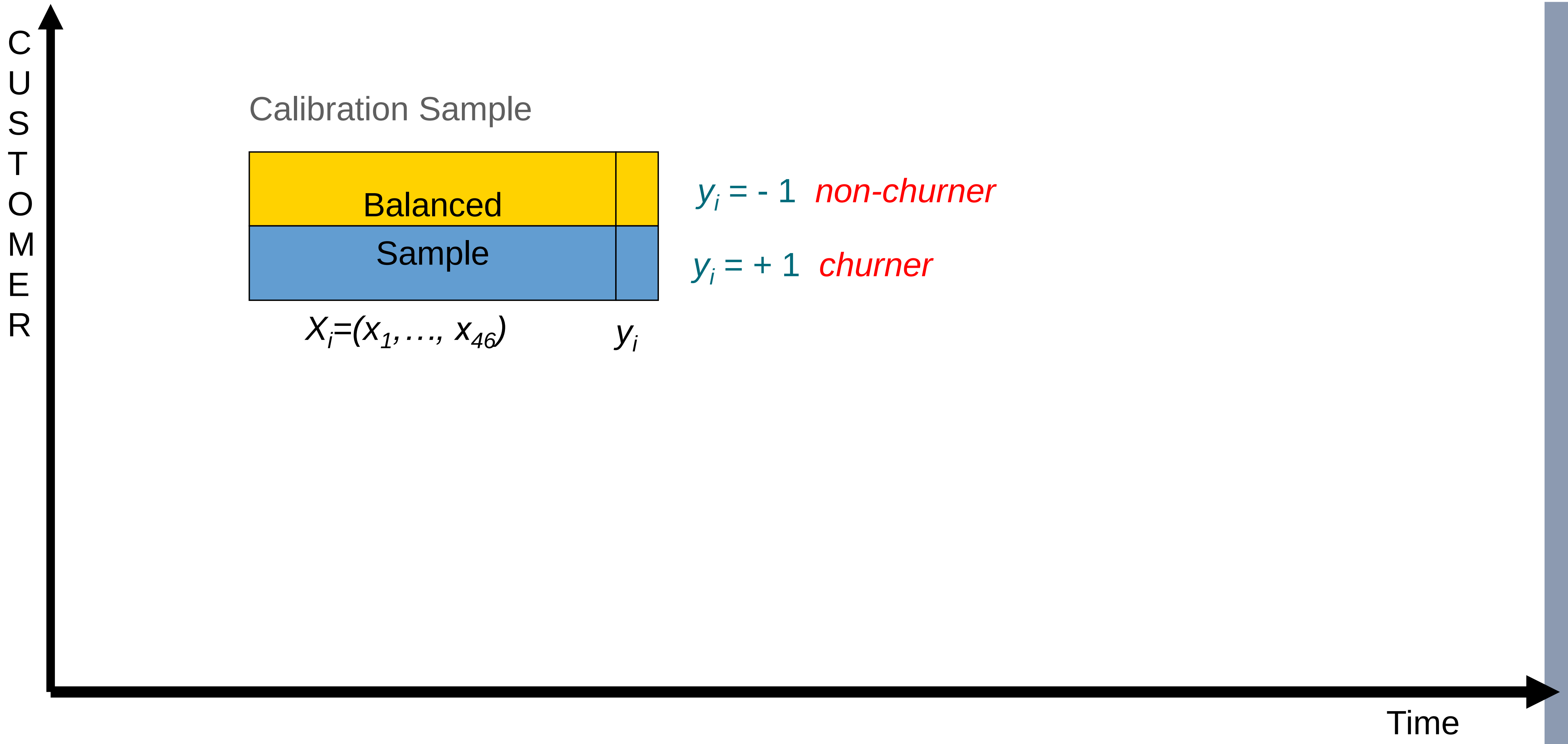


http://robjhyndman.com/hyndsight/quantile-forecasts-in-r/

How to split the data

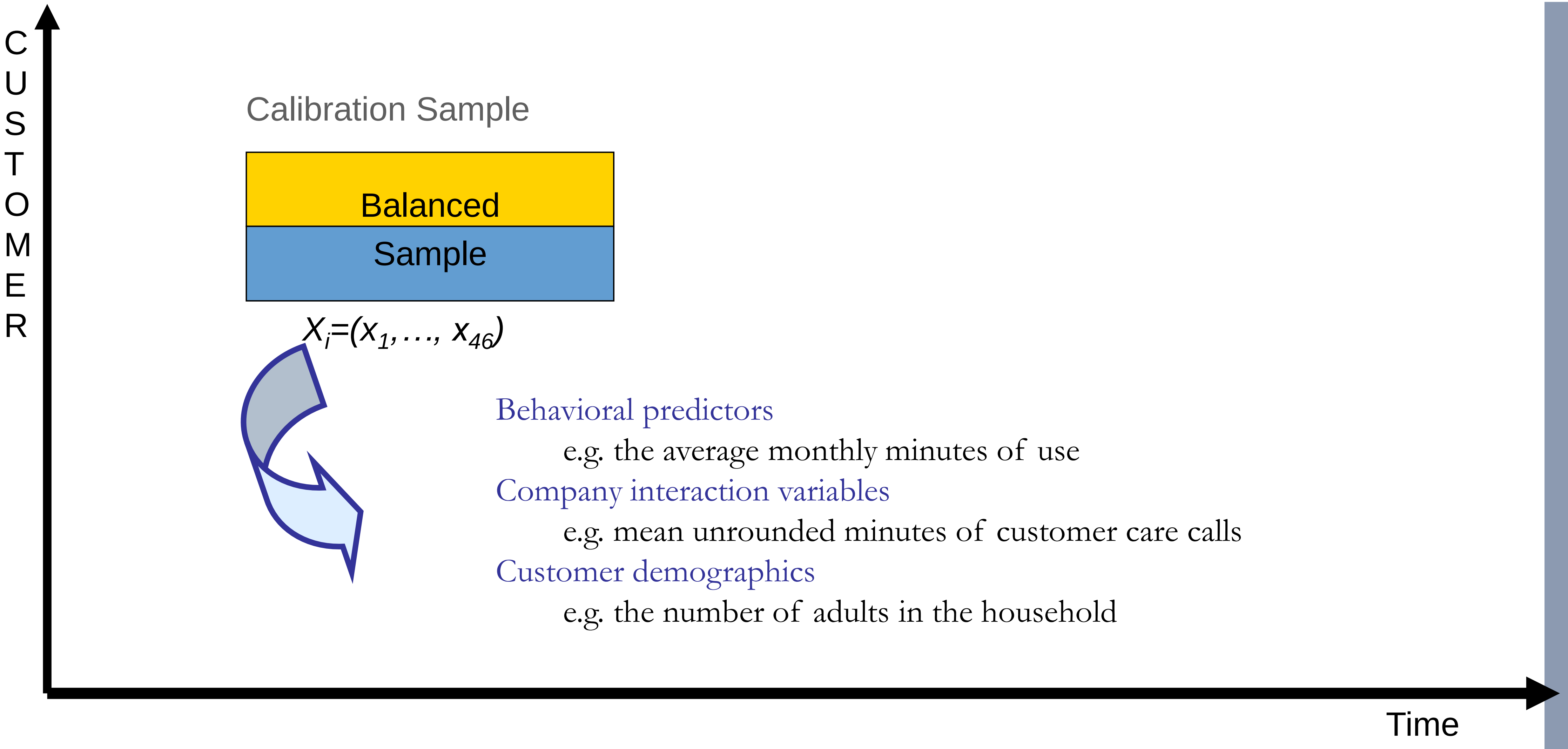
- Cross-sectional split
- Longitudinal/Temporal split
- Mix approach

Example: churn prediction



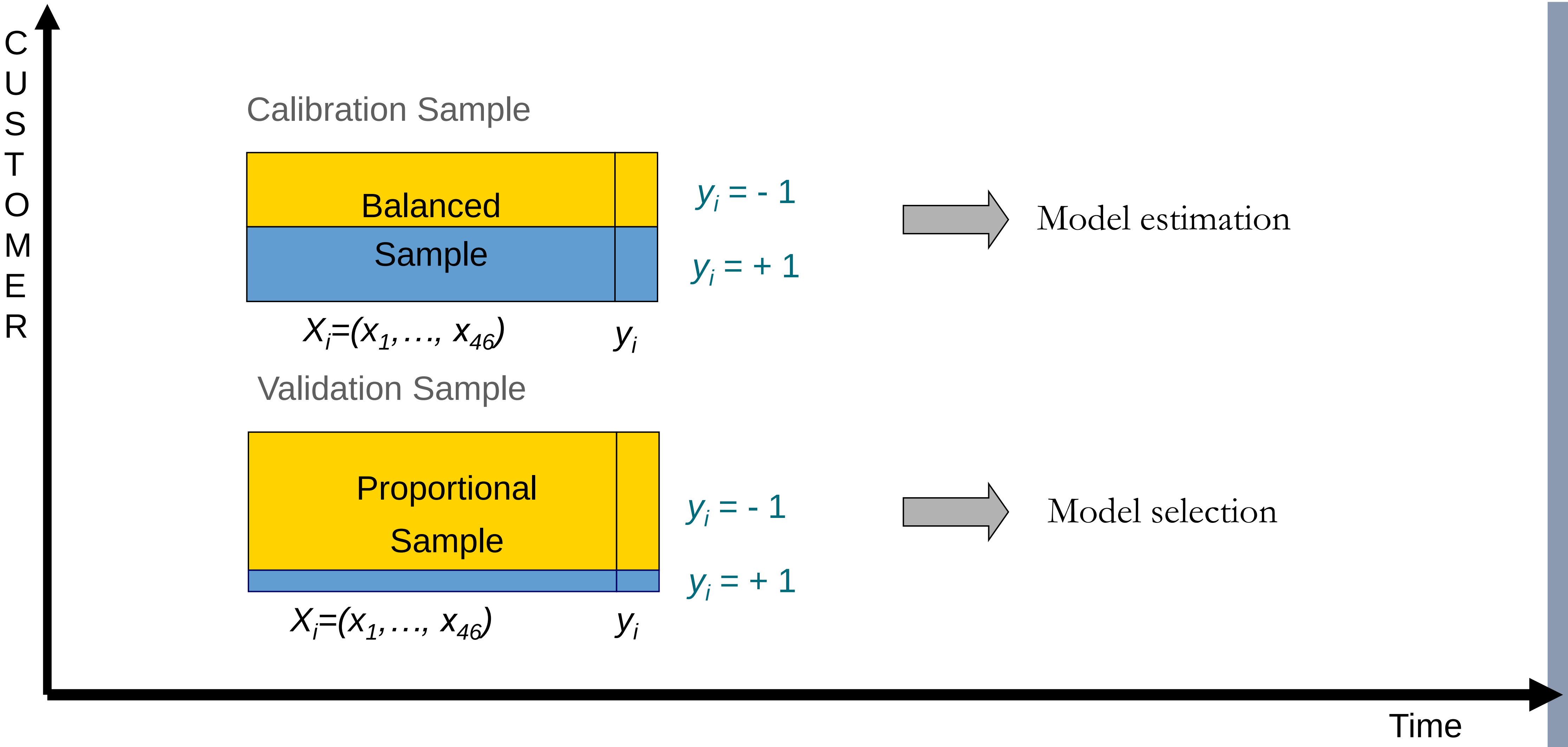
Lemmens and Croux (2006)

Example: churn prediction



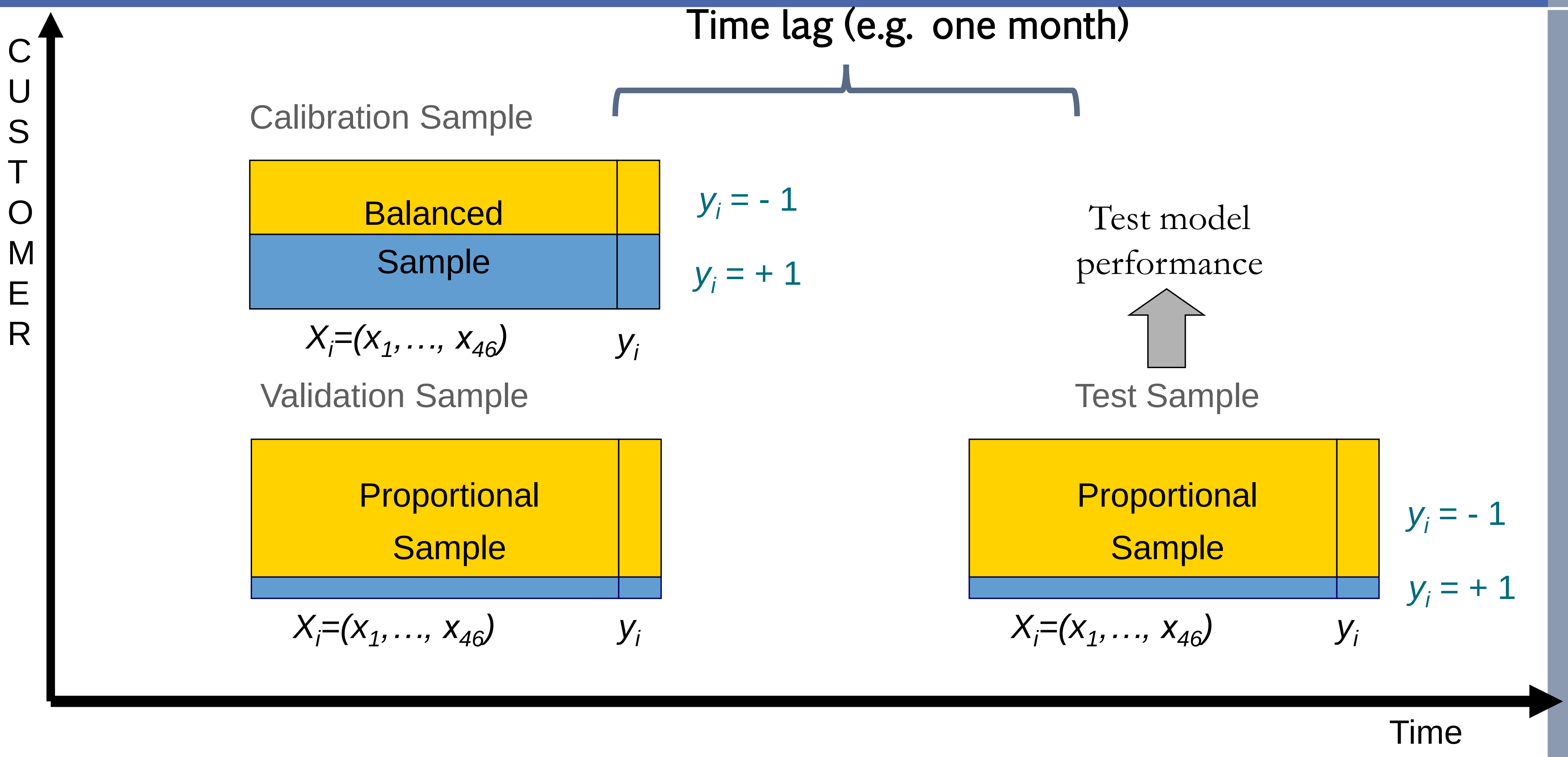
Lemmens and Croux (2006)

Example: churn prediction



Lemmens and Croux (2006)

Example: churn prediction



Lemmens and Croux (2006)

Bias due to Balanced Sampling

- Balanced versus proportional calibration sample
- Overestimation of the number of ones
- Several bias correction methods exist (see e.g. Cosslett 1993; Donkers et al. 2003; Franses and Paap 2001, p.73-75; Imbens and Lancaster 1996; King and Zeng 2001a,b; Scott and Wild 1997).

The Bias Correction Methods

- The weighting correction:
  - Based on prior beliefs about the actual proportion of ones, we attach weights to observations of a balanced calibration sample.

```
mylogit=glm(factor(y)~
              -1+.,as.data.frame(my.data.train),family=binomial(link="logit"),weights=ww)
```

- The intercept correction:
  - Take a non-zero cut-off value such that the proportion of predicted churners in the calibration sample equals the actual a priori proportion of churners.

```
percentage.ones=.2
cutoff=quantile(predict(mylogit,type="link",as.data.frame(my.data.test)),1-percentage.ones)
predicted.class=(predict(mylogit,type="link",as.data.frame(my.data.test))-cutoff)>=0
mean(predicted.class)
```

Example: new product forecast

- The issue of *pre-launch* forecasts: defining a fair prediction context

For forecasts made before the first international launch, no information on the actual adoption of the new product is available yet. If we denote $\bar{t}_{ij_0}$ as the year when a new product j_0 is introduced in country i, the *estimation sample* to forecast penetration of product j_0 in country i at prediction horizon h only includes the penetration data of other products available prior to $\bar{t}_{ij_0}$. In other words, we cut our data sample according to the calendar time at $\bar{t}_{ij_0}$. All data corresponding to the years after product j_0 has been launched in country i do not belong in the estimation sample.

Lemmens et al. (2012)

Example: new product forecast

For subsequent entries (all countries entered after the first country entered), some information on the actual adoption of the product became available. More specifically, the *estimation sample* to predict the penetration of a new product j_0 in a subsequent entry i' at prediction horizon h also includes the penetration data on product j_0 in previously-entered countries up to the introduction time in the focal country i'. We use the penetration of product j_0 in the focal country for all available years as a *hold-out sample*. Thus, for each product-country combination, we construct a different estimation and hold-out sample, divided according to calendar time. This framework replicates the data context practitioners face when making prelaunch forecasts.

Lemmens et al. (2012)

Example: new product forecast

Table 6
Prelaunch mean absolute forecast errors (MAD) per product in the hold-out sample, for various prediction horizons from $h=1$ to $h=5$ for all models[a].

DVD Players

	$h=1$	$h=2$	$h=3$	$h=4$	$h=5$
Parametric models					
Single-country segments	0.78	1.14	1.49	1.83	2.15
A priori-defined segments	0.52	0.85	1.30	1.80	2.45
Static segments	0.43	0.70	1.20	1.76	2.55
Dynamic segments	0.37	0.71	1.22	1.77	2.44
Semiparametric models					
Single-country segments	0.68	1.00	1.32	1.68	2.04
A priori-defined segments	0.27	0.45	0.75	1.07	1.48
Static segments	0.28	0.45	0.72	1.05	1.49
Dynamic segments	**0.15**	**0.23**	**0.32**	**0.48**	**0.64**

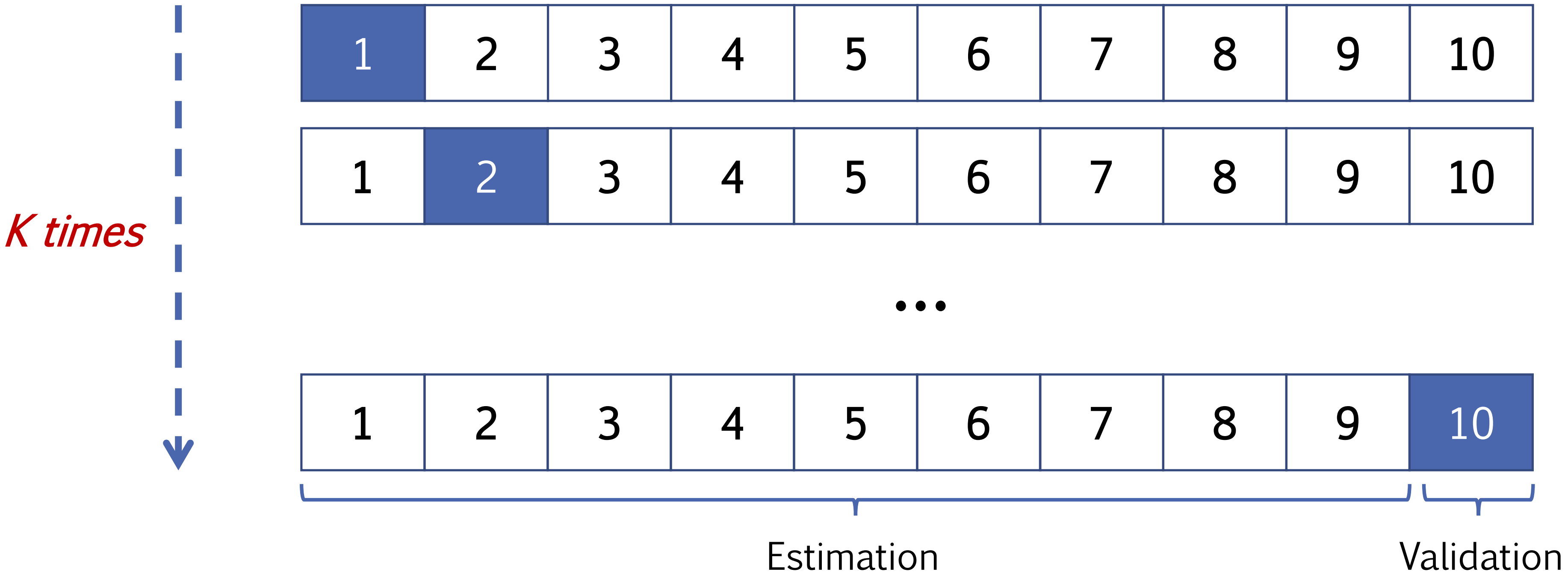
Lemmens et al. (2012)

Pros and cons

- Pros
  - No parametric or theoretical assumptions
  - Accurate if enough data is available
  - Simple to implement
- Cons
  - Be careful for contamination
  - We loose information
- Alternative approaches when the number of observations is limited: *cross-validation*
  - 10 fold cross-validation
  - Leave one out

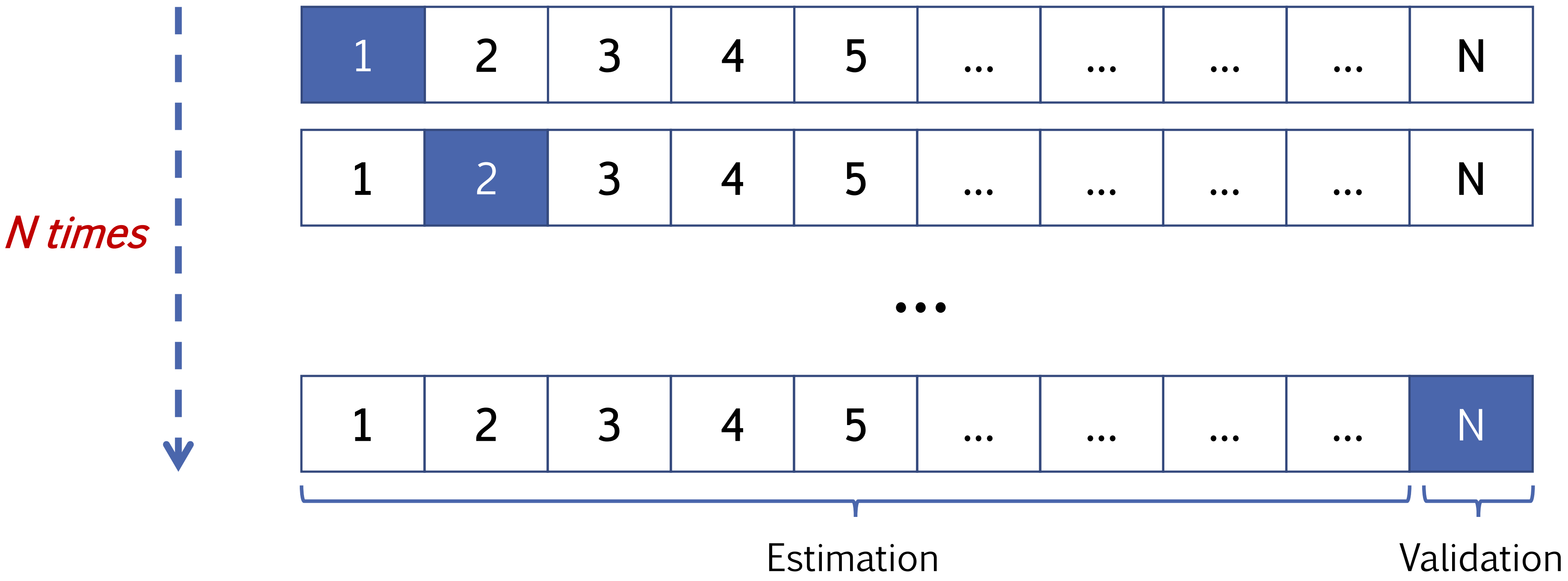
Cross-validation

› K-fold cross-validation, e.g. K = 10
  - Split the sample in K parts of equal size (minimum 1 obs.)
  - Estimate the model K times on K-1 parts
    › Sample size of one estimation sample = N - K
  - Validate the model of the remaining part



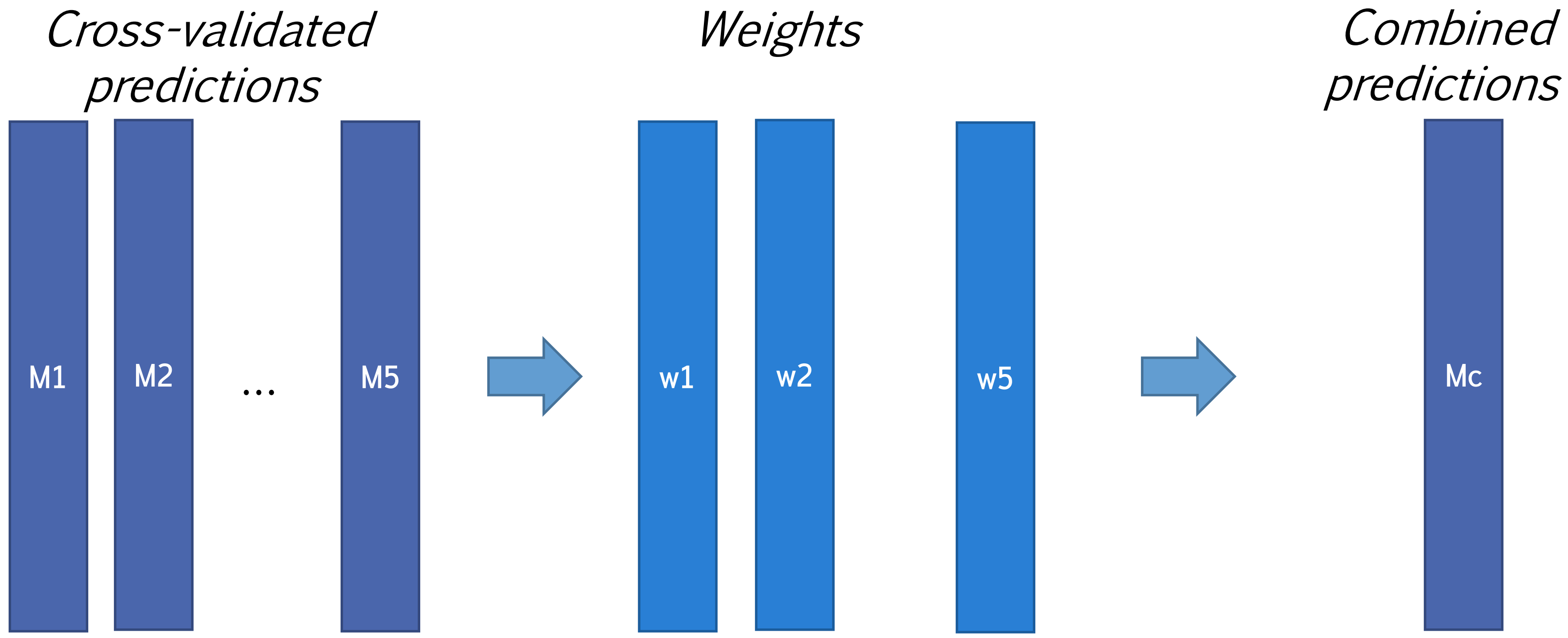
Cross-validation

- Leave-one-out
  - Same as K-fold cross-validation, where each part contains one observation
  - Estimate the model N times for a sample size of N
    - Sample size of one estimation sample = N - 1



Cross-validation for model averaging

- Calculate cross-validated predictions for each model
- Estimate model weights that optimize a given criteria
- Compute the weighted average of holdout predictions



The choice of benchmarks

Benchmark selection

- The model comparison should allow the reader to explain why the chosen approach performs better
- An approach often provides improvement on multiple dimensions,
  - E.g. adding heterogeneity and dynamics
- Always start with providing a table that shows where your contribution is.
- For model comparison, use by preference a full factorial design.
- If not possible, make sure to define the best fractional factorial design

Benchmark selection

› Full factorial design or fractional factorial design

Table 1
Model comparison.

Models	Semiparametric vs. parametric response model	Multi-country vs.single-country segments	Model-based vs. a priori-defined segmentation	Dynamic vs. static segments
Benchmarks				
Single-country segments	Parametric	Single-country	A priori	Static
	Semiparametric	Single-country	A priori	Static
A priori-defined segments	Parametric	Multi-country	A priori	Static
(geographic regions)	Semiparametric	Multi-country	A priori	Static
Static segments	Parametric	Multi-country	Model-based	Static
	Semiparametric	Multi-country	Model-based	Static
Dynamic segments	Parametric	Multi-country	Model-based	Dynamic
Our proposal				
Dynamic segments	Semiparametric	Multi-country	Model-based	Dynamic

Performance criteria

Performance criteria

- The loss assigns a cost/penalty to a prediction error
  - Regret associated with a suboptimal prediction/decision

- Which errors do we want to avoid/penalize?

Aligning the Loss to the Managerial Objectives

- Current practice:
  - Mismatch loss function (in-sample estimation) and performance evaluation (out-of-sample).
  - Such mismatch leads to suboptimal model selection and predictions (Engle 1993, Granger 1993)
- Rare exceptions:
  - Blattberg and George (1992): optimize manufacturers' prices by estimating price sensitivity using a profit-based loss function
  - Bult (1993) and Bult and Wittink (1996): loss function that account for the asymmetry in the cost of mistargeting mailing
  - Bayesian decision analysis (Rossi and Allenby, 2003; Gilbride, Lenk and Brazell, 2008)

Significance of the performance measures

- Use bootstrapping as seen before to generate multiple statistics of interest
- The bootstrap uses computer simulation but, instead of drawing observations from a hypothetical world, the bootstrap draws observations only from your own sample (not a hypothetical world)
- It makes no assumptions about the underlying distribution in the population.
- The standard error is the amount of variability in the statistic if you could take repeated samples of size n.
- How do you take repeated samples of size n from n observations? Sampling with replacement!
- Compute the variance across the simulated performance measures to determine significance

R code: significance of performance measures

```
S=100
train.index=1:floor(n/2)
test.index=(floor(n/2)+1):n
my.data.train=my.data[train.index,]
my.data.test=my.data[test.index,]
pred.test=array(NA,c(nrow(my.data.test),S))
boottop=bootgini=array(NA,c(S,1))

for (s in 1:S)
{
  bootstrap.index=sample(1:nrow(my.data.train),nrow(my.data.train),replace=TRUE)
  my.boot.data=my.data.train[bootstrap.index,]
  mylogit=glm(factor(y)~-1+.,as.data.frame(my.boot.data),family=binomial(link="logit"))
  pred.test[,s]=predict(mylogit,type="response",as.data.frame(my.data.test))
}
average.pred.test=apply(pred.test,1,mean)
topf(y=my.data.test$y,p=average.pred.test,share=.1)
ginif(y=my.data.test$y,p=average.pred.test)

for (s in 1:S)
{
  boottop[s]=topf(y=my.data.test$y,p=pred.test[,s],share=.1)
  bootgini[s]=ginif(y=my.data.test$y,p=pred.test[,s])
}
sd(boottop) ; sd(bootgini)
```

Thank you for your participation!
Questions?