Some statistical considerations on regression modeling with focus on predictive modelling

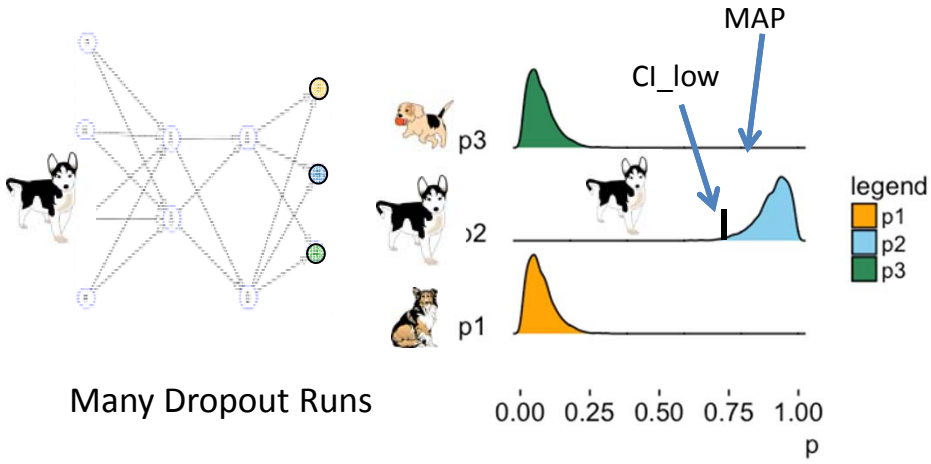
Beate Sick

Outline

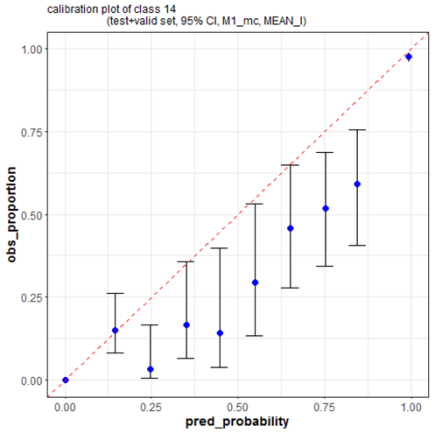
- Motivation
- Descriptive modeling revisited
- Causal modeling revisited
- Predictive modeling
  - Evaluation
  - Calibration
  - Shrinkage
- Attempts to explain why LS predictions are not calibrated
- Appendix: Regression with errors in variables

What has triggered this talk?

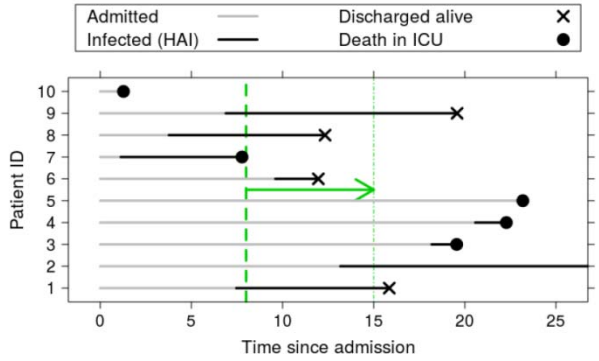
Probabilistic CNN predictions @ IDP

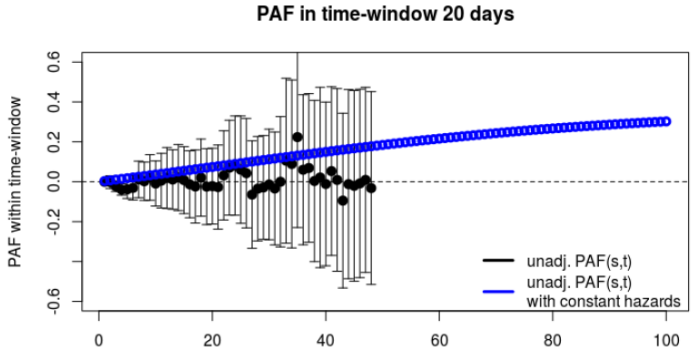


Many Dropout Runs



Summer School on preditive time-to-event models @EBPI





Who has triggered this talk?
Thanks for stimulating discussions and lots of input!

Probabilistic CNN predictions @ IDP

Oliver Dürr

Elvis Murina

Regression and predictive modelling @EBPI

Eva Furrer

Leo Held

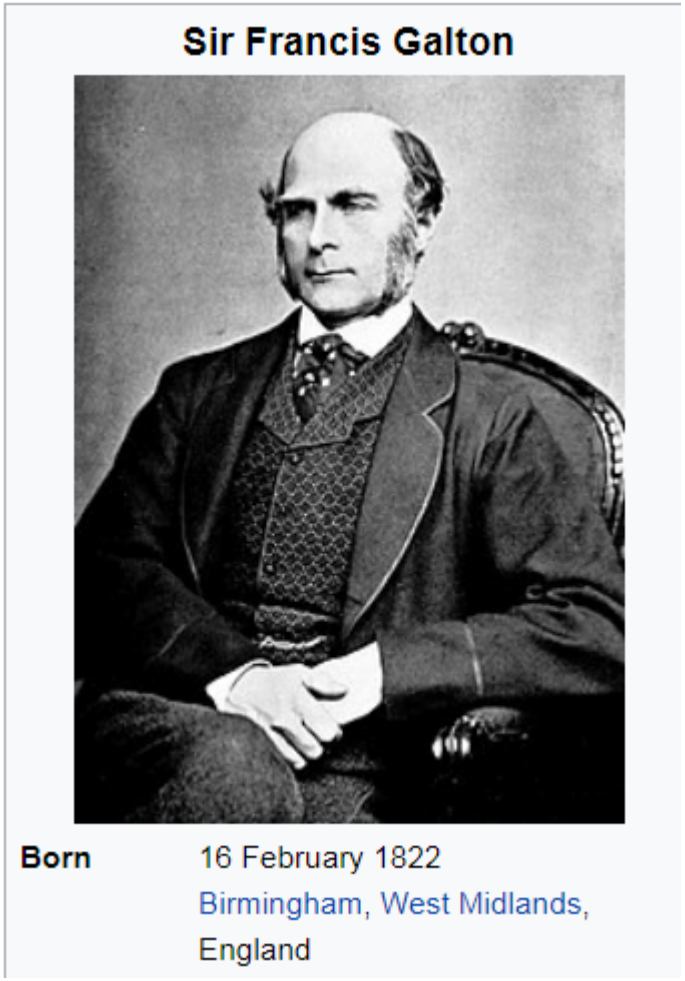
Burkhardt Seifert

Where does the term "regression" come from?

The concept of regression comes from genetics and was popularized by Sir Francis Galton during the late 19th century with the publication of *Regression towards mediocrity in hereditary stature*.

Galton observed that extreme characteristics (e.g., height) in parents are not passed on completely to their offspring. Rather, the characteristics in the offspring *regress* towards a *mediocre* point (a point which has since been identified as the mean).

Source: https://en.wikipedia.org/wiki/Francis_Galton



For what purpose do we develop a statistical model?

Statistical Science
2010, Vol. 25, No. 3, 289–310
DOI: 10.1214/10-STS330

To Explain or to Predict?

Galit Shmueli

Galit Shmueli is Tsing Hua Distinguished Professor at the Institute of Service Science, and Director of the Center for Service Innovation & Analytics at the College of Technology Management, National Tsing Hua University, Taiwan.

- **Description:**
  Describe data by a statistical model.
- **Explanation:**
  Search for the "true" model to understand and causally explain the relationships between variables and to plan for interventions.
- **Prediction:**
  Use model to make reliable predictions.

Linear regression – the mother of all statistical models
as used in descriptive modelling

Model for the conditional probability distribution

CPD: $Y_{X_i} = (Y|X_i) \sim N(\mu_{x_i}, \sigma^2)$

$Y_x \in \mathbb{R}$

$\mu_x \in \mathbb{R}$

μ_x is given by the model

σ^2 is independent of the predictor values

$$y_i = \beta_0 + \beta_1 \cdot x_{i1} + \varepsilon_i$$

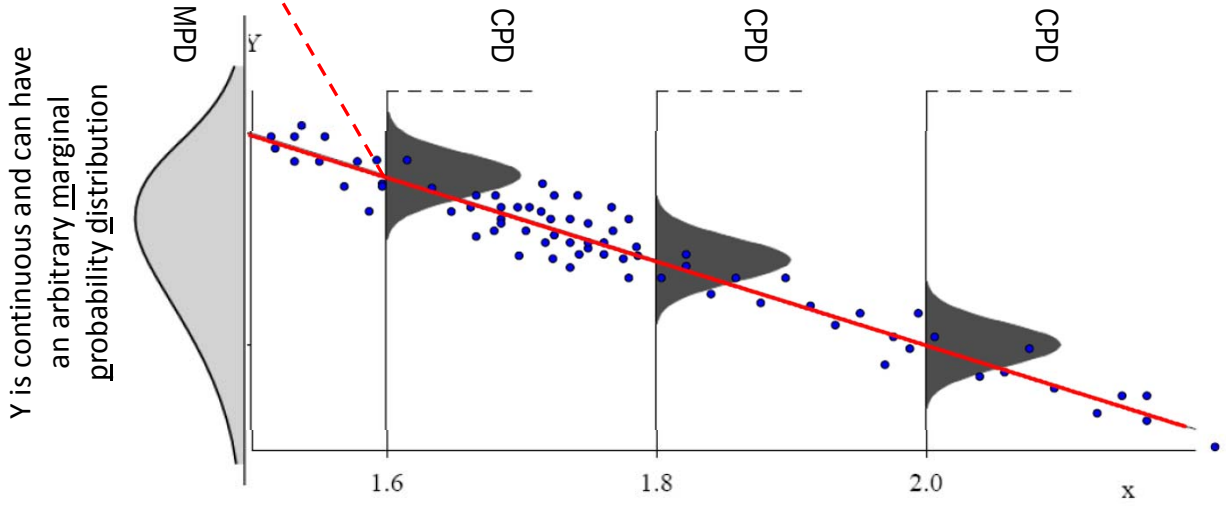
$$E(Y_{X_i}) = \mu_{x_i} = (\mu|X=x_i) = \beta_0 + \beta_1 \cdot x_{i1}$$

$$Var(Y_{X_i}) = Var(Y|X_i) = Var(\varepsilon_i) = \sigma^2$$

$$\varepsilon_i \text{ i.i.d.} \sim N(0, \sigma^2)$$

$$Y \sim V^{contiuous}_{arbirary}$$

$$(Y|X_i) \sim N(\mu_{x_i}, \sigma^2)$$



Linear regression: interpretation of coefficient

as used in descriptive modelling

$$y_i = \beta_0 + \beta_1 \cdot x_{i1} + \ldots + \beta_p \cdot x_{ip} + \varepsilon_i \quad , \quad \varepsilon_i \sim N\left(0, \sigma^2\right)$$

$$\mathbf{y} = \mathbf{X} \cdot \boldsymbol{\beta} + \boldsymbol{\varepsilon} \qquad \text{with } \mathbf{y} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}, \mathbf{X} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ 1 & x_{21} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{pmatrix}, \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_0 \\ \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

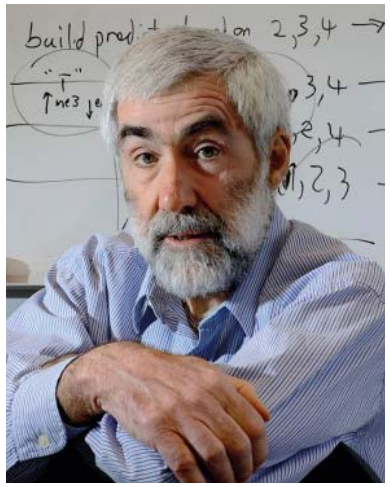
The coefficient β_k gives the change of the outcome y, given the explanatory variable x_k is increased by one unit and all other variables are held constant.

$$\beta_k = y_{x_k+1} - y_{x_k} = \Delta y_{x_k -> x_{k+1}}$$

Descriptive modeling has a long tradition in statistics

"Descriptive modelling is aimed at summarizing or representing given data in a compact manner. …the reliance on an underlying causal theory is absent"

Source: Shmueli 2010



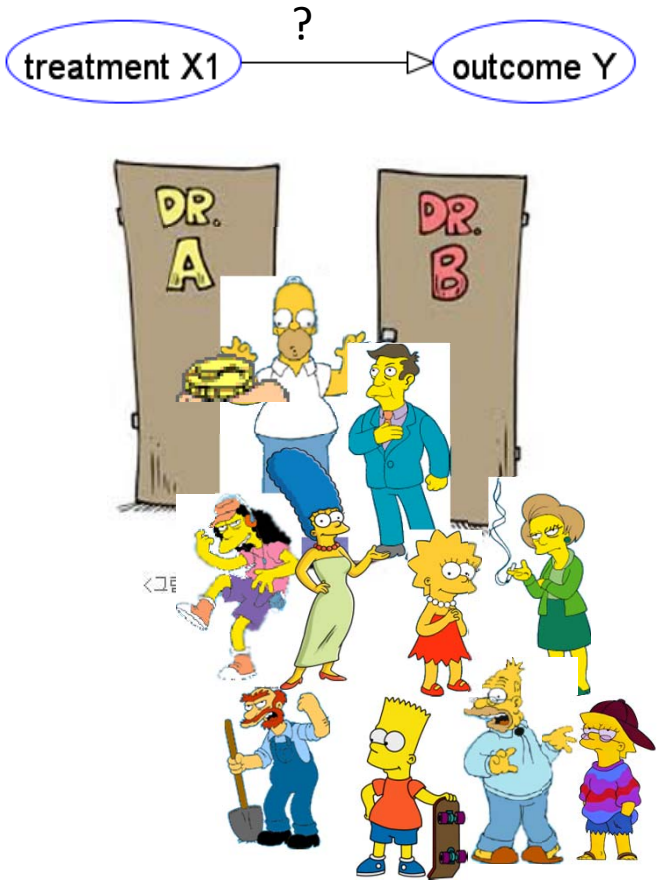
"Considerations of causality should be treated as they have always been in statistics: preferably not at all;

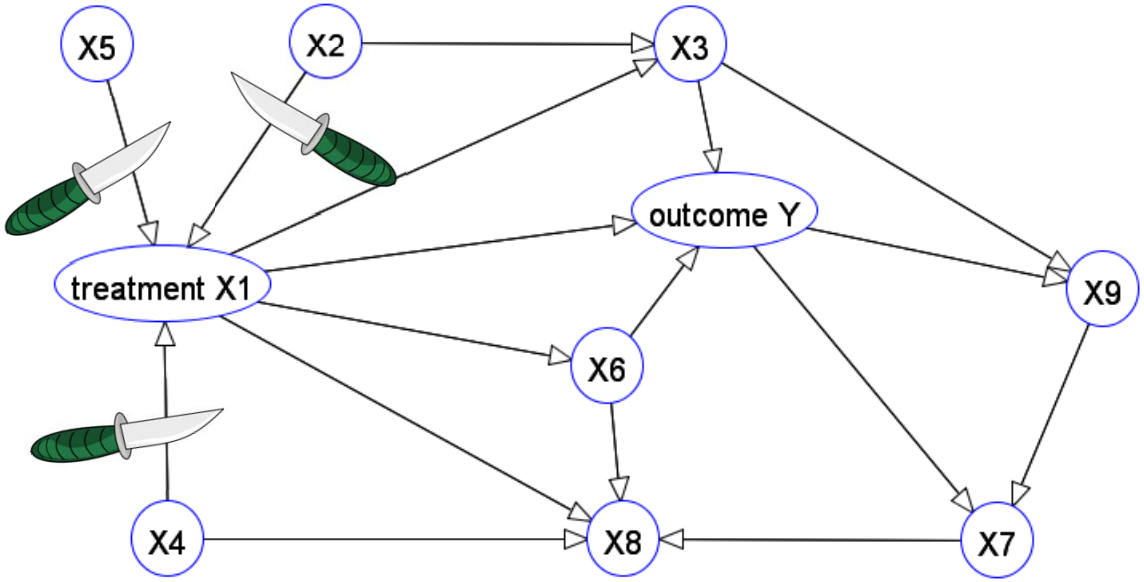
But if necessary than with great care"

Terry Speed, president of the Biometric Society 1994-95

Especially we cannot use coefficients in linear regression model to predict how the outcome y would change if we make an intervention and increase a certain predictor by one unit (we make no intervention on all other predictors neither do we control them).

Sidetrack: Causal effects are best derived from randomized trials





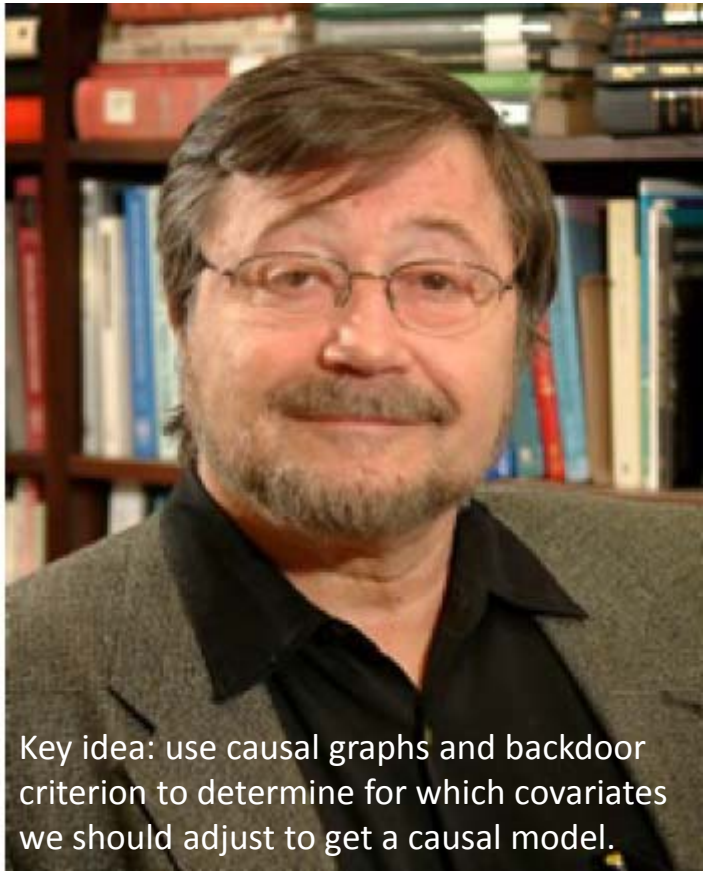
Treatment is assigned randomly

→ differences of the outcome between both treatment groups is due to the treatment:

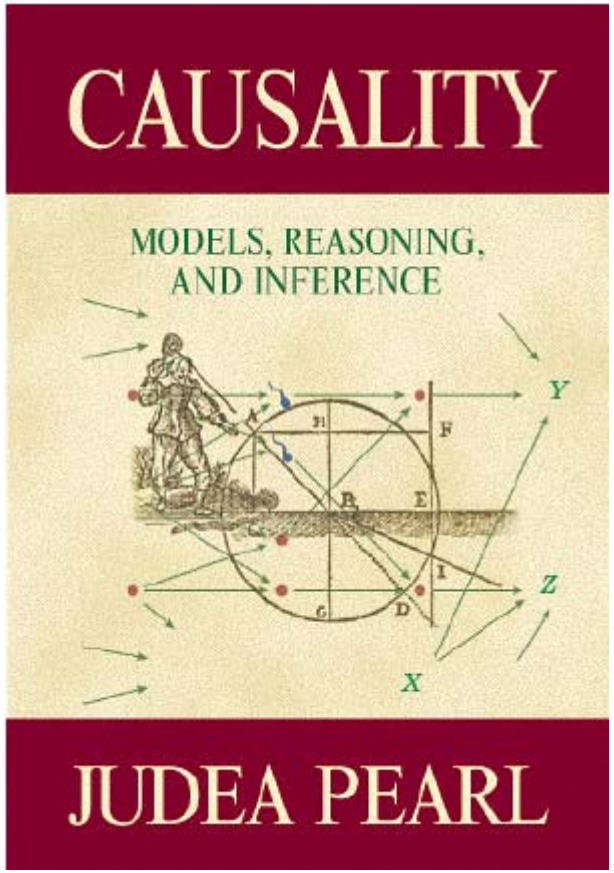
causal effect = intervention effect

→ appropirate regression model: $outcome \sim treatment$

Judea Pearl introduced causal reasoning into statistical modeling of observational data



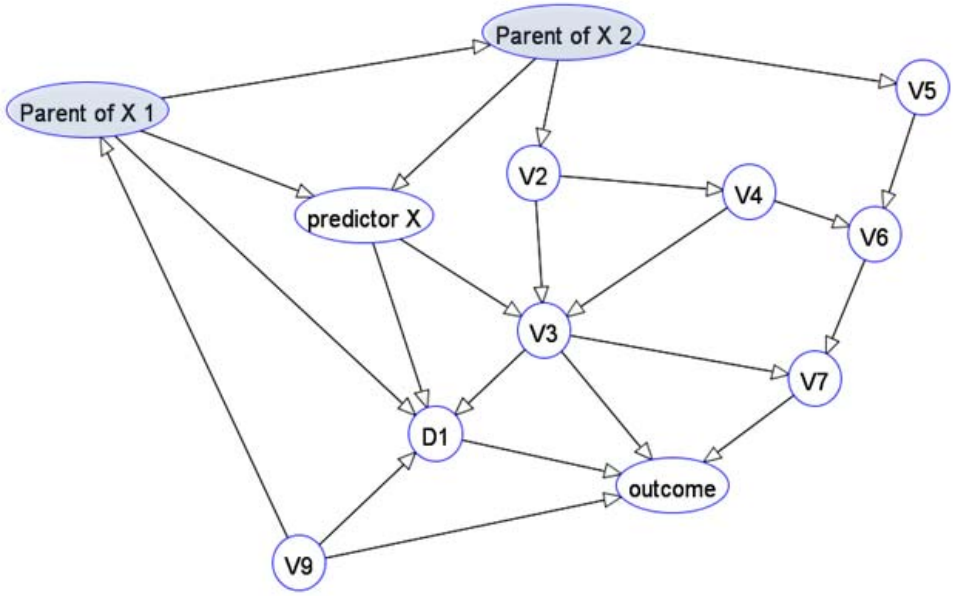
Key idea: use causal graphs and backdoor criterion to determine for which covariates we should adjust to get a causal model.



ACM Turing Award 2011: “For fundamental contributions to artificial intelligence through the development of a calculus for probabilistic and causal reasoning."

Sidetrack: Causal effects can also be derived from observational data when using Pearl's backdoor criterion is fulfilled

When can the regression coefficient in a model be interpreted as causal effect?



We need to adjust with an appropriate **set S_B** of covariates V_i (e.g. all parents of X) which would be sufficient to **close all backdoor paths** from intervention X to the outcome Y

$$\text{outcome} \sim \text{predictor} + \sum_{V_i \in S_B} V_i \qquad \text{outcome} \sim \text{predictor} + \sum \text{parents(predictor)}$$

Also statisticians picked up causality based on Pearl's ideas

P. Bühlman (ETH): "Pure regression is intrinsically the wrong tool"

(to understand causal relationships between predictors and outcome and to plan interventions based on observational data)"

Regression – the "statistical workhorse": the wrong approach

we could use linear model (fitted from n observational data)

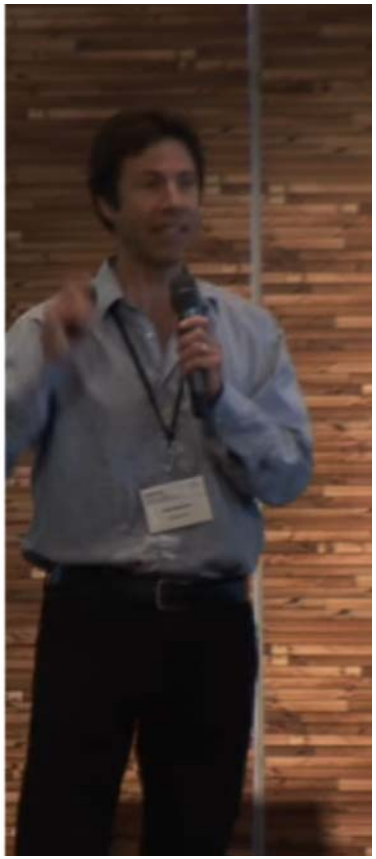
$$Y = \sum_{j=1}^{p} \beta_j X_j + \varepsilon,$$

$$\mathrm{Var}(X_j) \equiv 1 \text{ for all } j$$

$|\beta_j|$ measures the effect of variable X_j in terms of "association"

i.e. change of Y as a function of X_j when keeping all other variables X_k fixed

$\rightsquigarrow$ not very realistic for intervention problem
if we change e.g. one gene, some others will also change
and these others are not (cannot be) kept fixed



https://www.youtube.com/watch?v=JBtxRUdmvx4

Retrospective vs Prospective modeling

- Descriptive modeling is retrospective:
  the model is used to describe the collected data.

- Explanatory modeling is retrospective – the model is used
  - to test an already existing set of hypotheses
  - to estimate the causal effect of a predictor on the outcome

  for both cases appropriate data that were collected for this purpose

- Predictive modeling is prospective (forward-looking):
  the model is constructed for predicting new observations.

Model assessment of retrospective models is done on data used to build the model.

Model assessment of prospective models is done on new data.

Let's simulate some data and split them in train and test set

```
n = 200 # number of observation
prop_train=0.25  # proportion of obs to use for train
p = 10  # number of predictors
error_sd = 2 # added noise
max_abs_coef = 2 # defines range of "slopes"
min_coef = 0.3  # we set "effect"=0 if slope<min_coef

# sample p coefficients between min and max
beta = matrix(runif(p, min=-max_abs_coef,
                    max=max_abs_coef), ncol=1)
# set beta to 0 if sampled beta was smaller 0.3
beta = ifelse(beta<0.3, 0, beta)

# data matrix:
x = matrix(rnorm(n*p), nrow=n, ncol=p)

# generate outcome y according to linear model plus error
y = x %*% beta + rnorm(n, sd=error_sd)

# built data frame holding predictors and outcome
dat_sim = data.frame(y, x)
head(dat_sim)
```

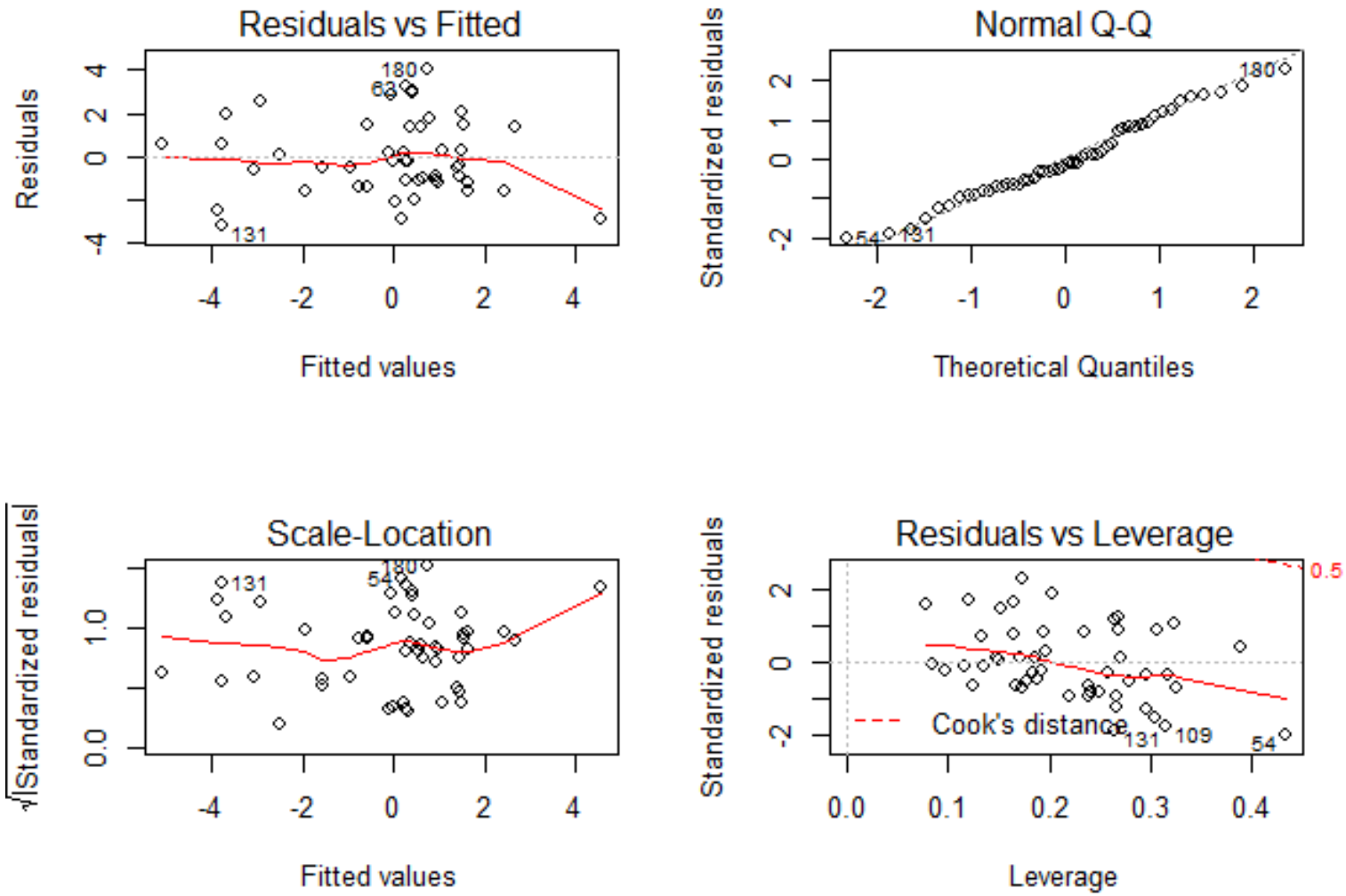
```
           y         X1          X2         X3         X4         X5         X6         X7         X8         X9        X10
 0.305239571  0.04266307 -1.81394212  1.1000002  0.68120622 -1.0160885 -1.3009742  1.5188241 -1.0362367  0.81158385  0.6602157
 0.762806963  0.01829321 -0.46685921  0.2346569 -2.66254636  0.1124149  1.8886849  1.3943780 -1.4605380 -0.19481145 -1.2991316
 0.006516694  0.97322521 -0.63294606  0.7620599 -0.47316238 -1.3423427 -0.9987358 -0.1026116  0.8962294 -0.33722534 -0.3553637
-6.344377597  2.00650835 -0.01336105  0.4250921  0.01623457  0.6719203 -0.9727473 -0.4315897  0.5360095 -0.19686322 -1.2474868
 4.633133787 -0.89475932 -0.03973259  1.2099400  0.73924955 -0.1493675 -0.9959355 -1.5872068 -0.4715230 -0.09685843  0.9034193
-1.085515170 -0.23540671 -0.12748278 -0.7399379  0.05949176 -0.8293857 -1.8514649 -1.0149369 -0.8859952 -0.64563866  1.6380714
...
```

```
# randomly split data in training and test
idx.tr = sample(1:nrow(dat_sim),
                as.integer(prop_train*nrow(dat_sim)))
train = dat_sim[idx.tr,]
test = dat_sim[-idx.tr,]
```

Let's fit a linear regression model for descriptive modelling

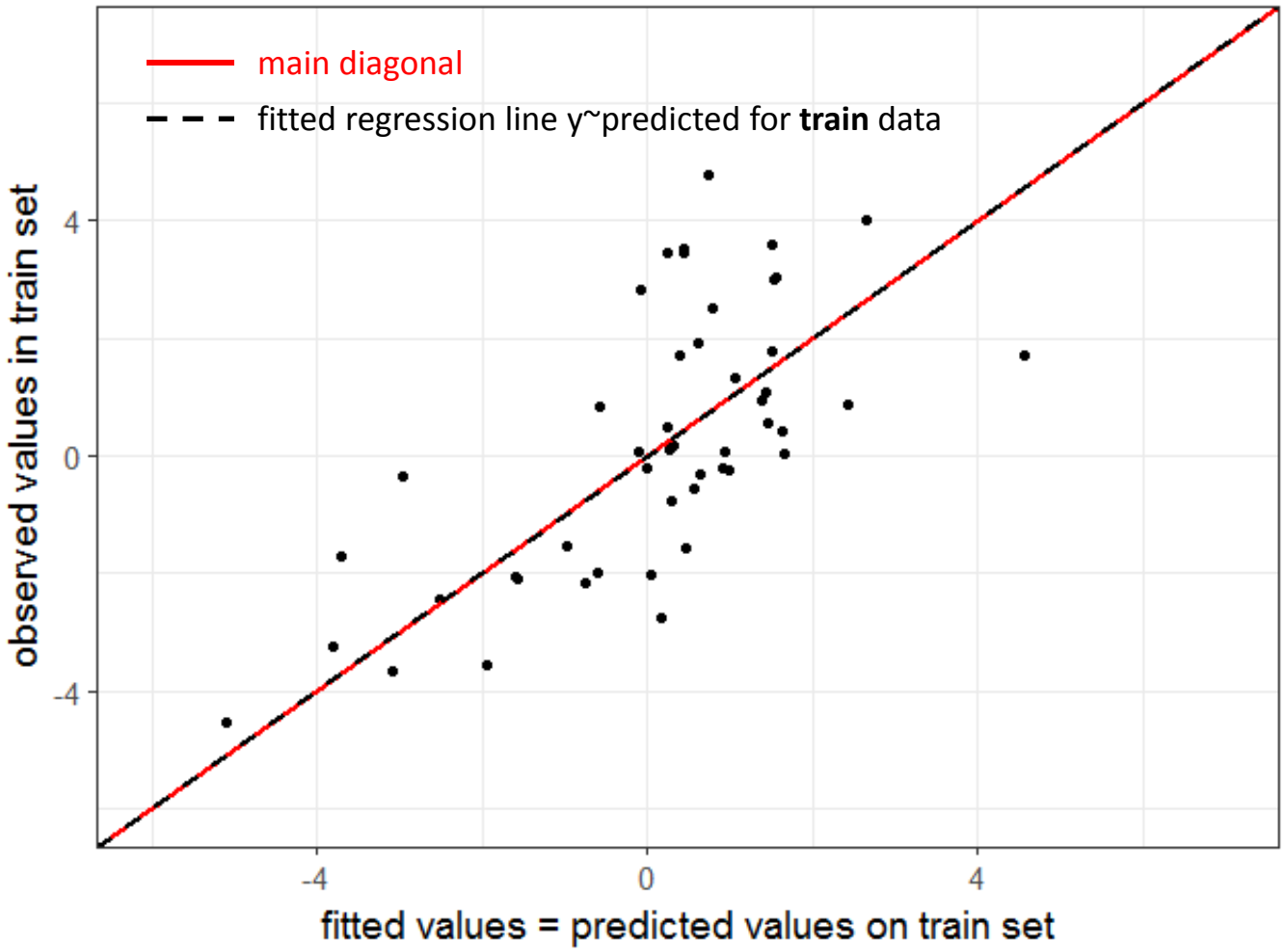
```
# now fit a regression model
fit_train = lm(y~. ,data=train )

# statistician's model check:
plot(fit_train)

# poor man's model check:
train$train_pred = predict(fit_train,
                           newdata=train)

# for unbiased fit we expect slope 1 for obs vs fitted
fit_train = lm(y ~ train_pred, data=train)

# check visually if fitted model is unbiased
ggplot(train , aes(x=train_pred, y=y)) +
  geom_point() +
  geom_abline(intercept=0, slope=1,
              linetype=1, colour='red', size=1) +
  geom_abline(intercept=coef(fit_train)[1],
              slope=coef(fit_train)[2],
              linetype=2, colour='black', size=1) +
  theme_bw(base_size = 14) +
  xlim(my_ymin,my_ymax) +
  ylim(my_ymin,my_ymax) +
  ylab("observed values in train set") +
  xlab("fitted values = predicted values on train set")
```

Statisticians descriptive model check: residual analysis



Residual plots look o.k., what is expected since true model is fitted to simulated data.

Poor man's descriptive model check: observed vs fitted

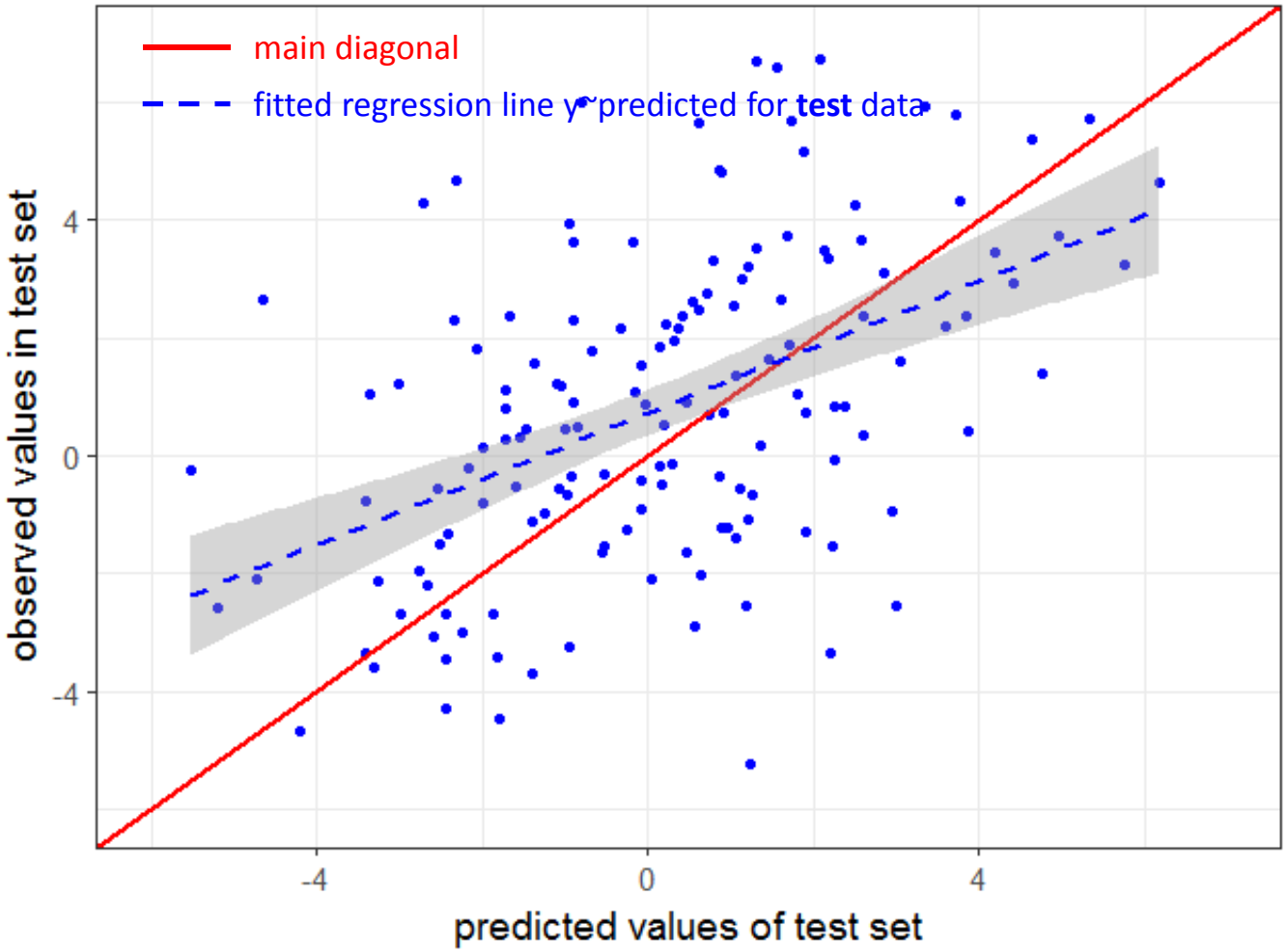


Slope of fitted line for observed vs fitted is 1 proving that linear regressions produces unbiased fitted values.

Let's use the regression model as predictive model

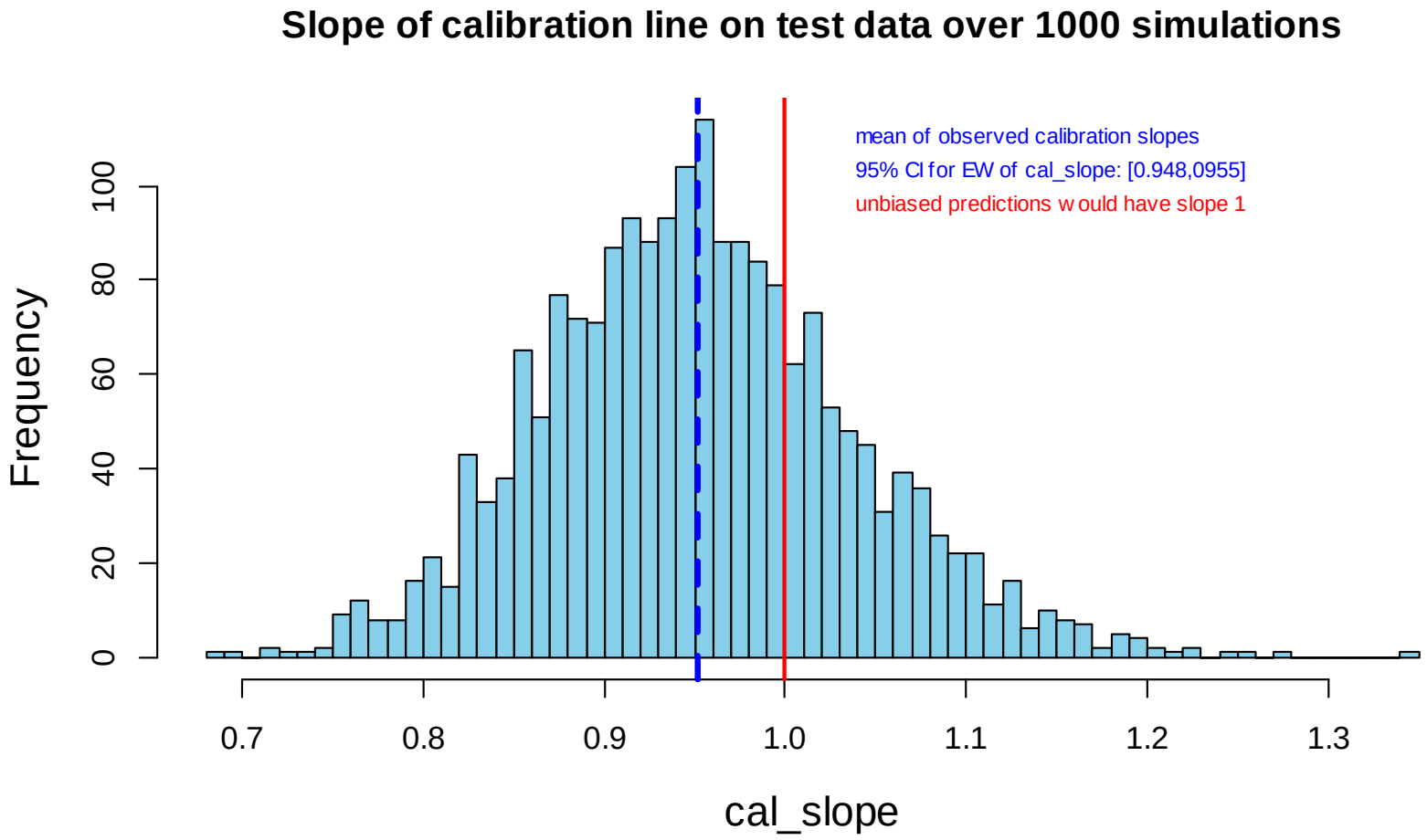
```
# fit a regression model on train data
fit_train = lm(y~. ,data=train )

## predict new test data with fit_train
test$test_pred = predict(fit_train, newdata=test)

# check if predictions on new data are unbiased
ggplot(test , aes(x=test_pred, y=y)) +
  geom_point(colour='blue') +
  geom_abline(intercept=0, slope=1,
              linetype=1, colour='red', size=1) +
  geom_smooth(method="lm",linetype=2, colour='blue' ) +
  theme_bw(base_size = 14) +
  xlim(my_ymin,my_ymax) +
  ylim(my_ymin,my_ymax) +
  ylab("observed values in test set") +
  xlab("predicted values of test set")
```

Let's check if predictions are unbiased: calibration plot



Slope of fitted line for observed vs predicted values is smaller that 1 meaning that linear regressions produces to extreme predicted values.

Are predictions always to extreme?



The simulation shows that on average linear regressions produces to extreme predicted values. But we have sample variation and the bias varies from run to run and sometimes the predictions are even not extreme enough.

In which cases are "naïve" predictions biased on new data?

If the regression fit (on training data) is not very good (small F statistics) indicating that the model rather overfits the data as it easily happens for

- lot of noise – large residual error
- small (training) data set
- small effects – small coefficients
- many predictors

→ see R simulation

Observations in R simulation:

- For p>=2 predictors and small effects we can observe this bias in the predictions
  (mean calibration slope b is significantly smaller than 1 when using the a model fitted on train to predict test data)
- For p=1 predictor (simple regression) we cannot estimate the expected value of the calibration slope b to show b <1
- The distribution of the simulated calibration slopes is not Normal but is rather compatible with a t-distribution with df=p

Houwelingen2001 p.25: "It might be expected that shrinkage in the regression setting only works if there are at least three covariates yielding four unknown parameters."

Shrinkage leads to calibrated predictions on new data

- Naïve Least Square Prediction uses the model fitted on train data also to predict new data:

$$\hat{\mathbf{y}}_{LS} = \hat{\boldsymbol{\alpha}}_{LS} + \mathbf{x} \cdot \hat{\boldsymbol{\beta}}_{LS}$$

- Since we observe that predictions on new data are on average to extreme, we need to shrink them towards the mean to get calibrated predictions

- This can be achieved by shrinking the coefficients in the regression model (w/o loss of generality we assume that all predictors are centered).

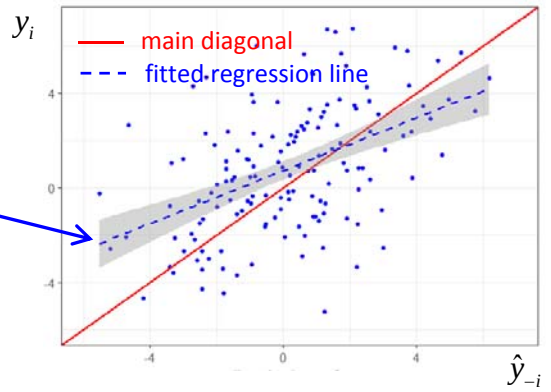
- Recalibrated prediction achieved by shrinking the coefficients with a shrinkage factor $c \in [0,1]$:

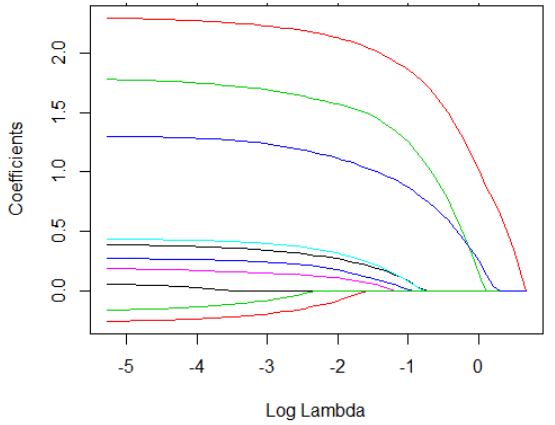
$$\hat{\mathbf{y}}_{\text{recalibrated}} = \hat{\boldsymbol{\alpha}}_{LS} + \mathbf{X} \cdot \left(c \cdot \hat{\boldsymbol{\beta}}_{LS} \right)$$

How to determine how much we need to shrink the coefficients?

- Based on the global F-statistics: $\hat{c} = \max\left(1 - \frac{1}{F}, 0\right)$
  (see Copas 1997)

- Or better through cross validation:
  - regress y_i on $\hat{y}_{-i}$ where $\hat{y}_{-i}$ is the naive LS prediction where the LS model was fitted while omitting the i-th observation: the slope of this regression is the estimated shrinkage factor c.
  - Use penalized regressions such as LASSO resulting in shrinked coefficients by applying a L1 penalty, where the tuning parameter λ is optimized for prediction performance and is determined by cv

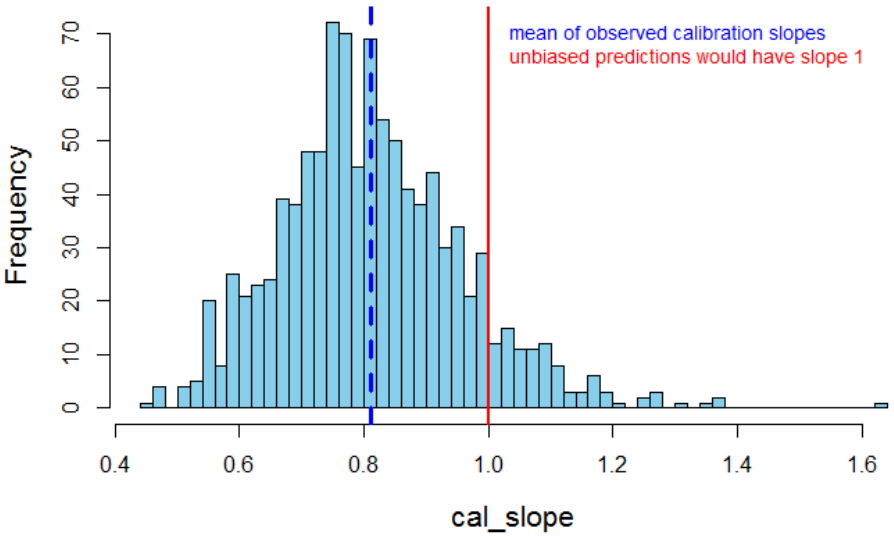




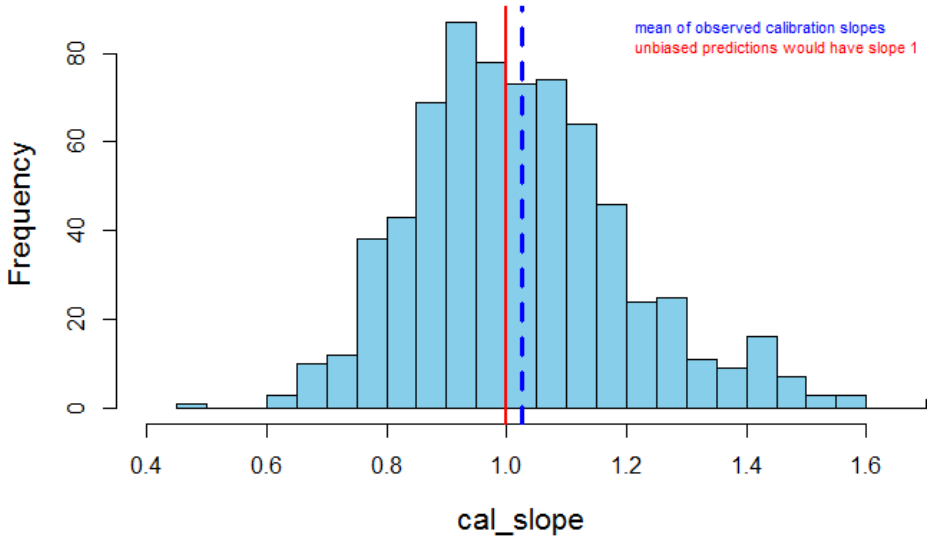
$$\hat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda\|\beta\|_1$$

Compare calibration of LS-lm versus cv-lasso model

(unshrinked) LS **lm** model



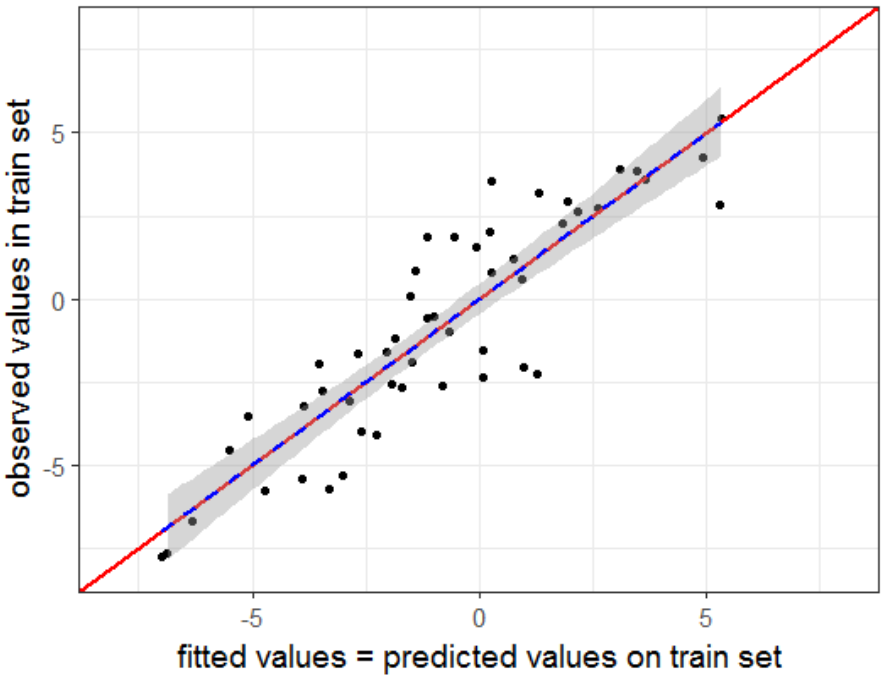
cv-shrinked **lasso** model

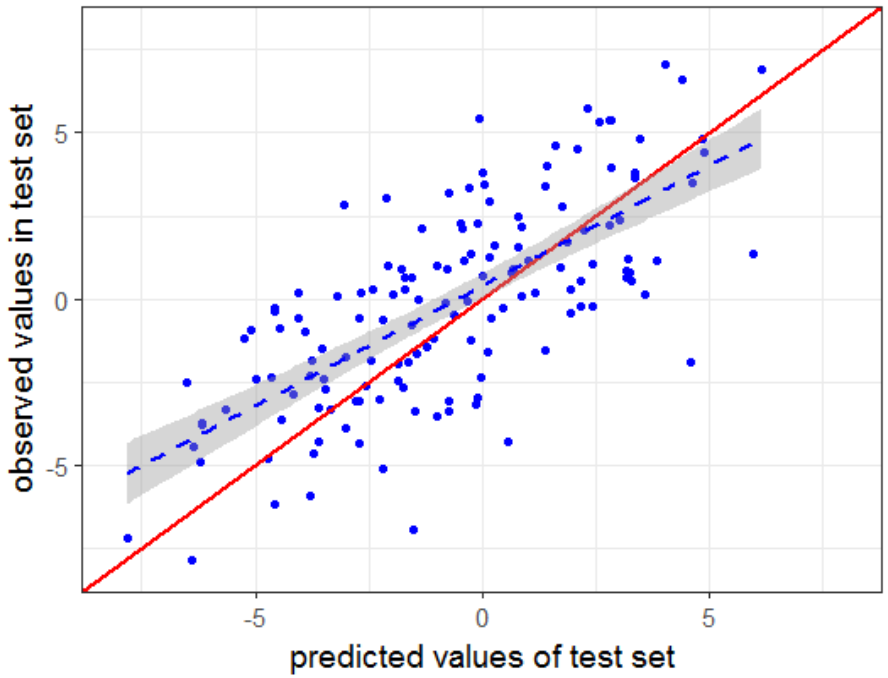


The linear model from the simulation (10 predictors) is better calibrated after Lasso shrinkage.

```
library(glmnet)
fit_lasso = glmnet(x=X, y=Y, alpha=1)
plot(fit_lasso, xvar = "lambda")
crossval =  cv.glmnet(x=X, y=Y)
plot(crossval)
opt_la = crossval$lambda.min
fit1 = glmnet(x=X, y=Y, alpha=1,
                  lambda=opt_la )
```

Calibration slope for train and test predictions





Calibration fit: $y = a + b \cdot \hat{y}$ where b is the calibration slope

Using the model that was fitted on training set for prediction we get:

always a calibration-slope = 1

when predicting training data

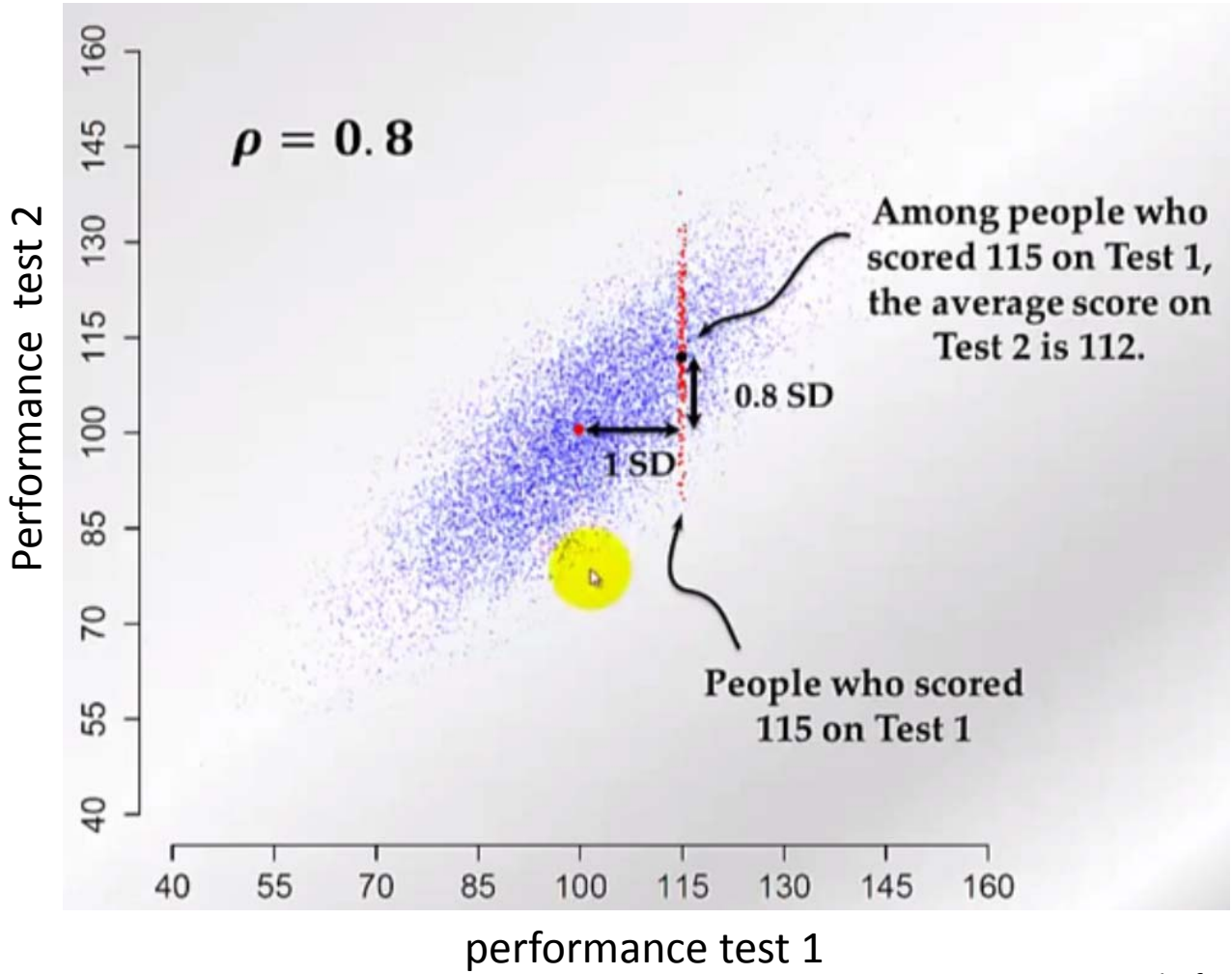
on average a calibration-slope < 1

when predicting test data

We want to understand / deduce, why naïve predictions on new data are biased

Sidetrack: Regression to the mean as step on the way

Consider a test-retest situation: Same people do twice a performance test:



Source: https://www.youtube.com/watch?v=aLv5cerjV0c

See appendix for a formal rational based on "errors in variables"

Sidetrack: Regression to the mean as step on the way

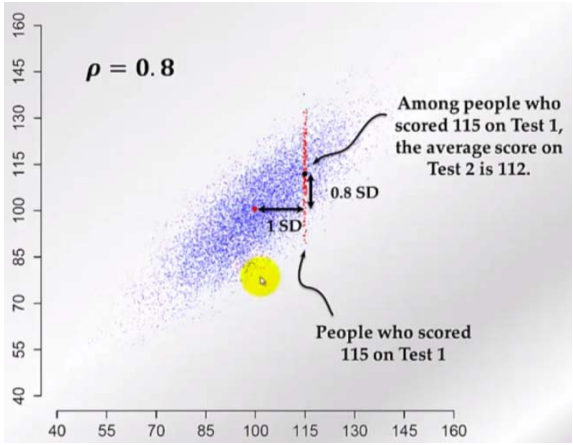
We look at 2 RV (random values) Y1 and Y2

- Both follow the same distribution → $\mathrm{E}(\mathrm{Y1}) = \mathrm{E}(\mathrm{Y2}) = \mu\,, \mathrm{Var}(\mathrm{Y1}) = \mathrm{Var}(\mathrm{Y2})$
- $\rho = |corr(Y1, Y2)| < 1$

Naïve (w/o shrinkage to the mean) we would expect: $E(Y2|Y1) = Y1$

However, we have regression to the mean (μ) meaning that the expected value of Y2 for all observations with a certain Y1 value is not also Y1 but closer to the overall mean μ.

$$E\left(Y2 \mid Y1\right) = \rho \cdot Y1 + (1-\rho) \cdot \mu$$



See appendix for a formal rational based on “errors in variables”

Sidetrack: Regression to the mean as step on the way
Proof by regression

- Y1, Y2 follow the same distribution → $\mu_{Y1} = \mu_{Y2} = \mu$, ${\sigma_{Y1}}^2 = {\sigma_{Y2}}^2 = \sigma^2$
- $\rho = |corr(Y1, Y2)| < 1$

We want to show: $E(Y2 \mid Y1) = \rho \cdot Y1 + (1-\rho) \cdot \mu$

Proof via regression:

$$Y2 = \beta_0 + \beta_1 \cdot Y1 + \varepsilon$$

$$E(Y2 \mid Y1) = \beta_0 + \beta_1 \cdot Y1$$

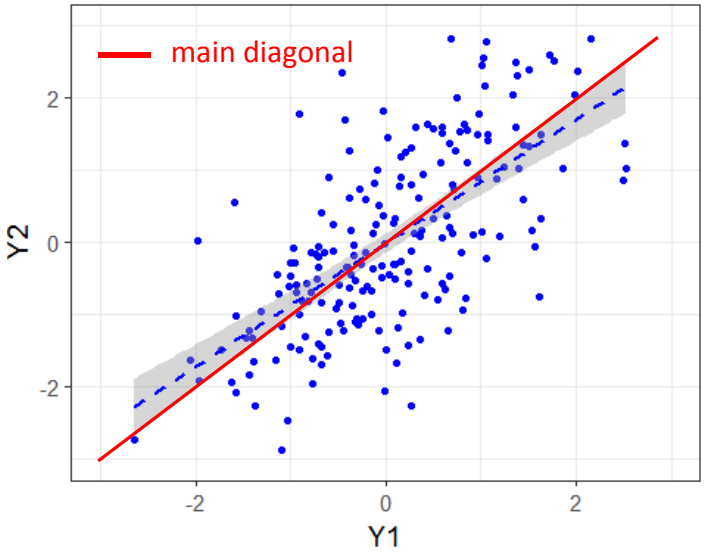
$$\beta_0 = \mu_{Y2} - \beta_1 \cdot \mu_{Y1}$$

$$\beta_1 = \rho \frac{\sigma_{Y2}}{\sigma_{Y1}}$$

$$E(Y2 \mid Y1) = \mu_{Y2} - \rho \frac{\sigma_{Y2}}{\sigma_{Y1}} \cdot \mu_{Y1} + \rho \frac{\sigma_{Y2}}{\sigma_{Y1}} \cdot Y1$$

$$E(Y2 \mid Y1) = \mu - \rho \frac{\sigma}{\sigma} \cdot \mu + \rho \frac{\sigma}{\sigma} \cdot Y1$$

$$E(Y2 \mid Y1) = \rho \cdot Y1 + (1-\rho) \cdot \mu$$



See appendix for a formal rational based on “errors in variables”

How to understand why naïve predictions are biased:
Regression to the mean as step on the way

In our simulation we have 10 predictors and we sampled 10 fix coefficients.

We have sampled 50 predictor vectors for the train data & simulated y from the true model:

y_train = x_train %*% beta + rnorm(n, sd=error_sd)

$$\mathbf{y}_{\text{train}} = \mathbf{X}\cdot\boldsymbol{\beta} + \boldsymbol{\varepsilon}' \quad \text{with } \varepsilon_i' \in \mathrm{N}\left(0,\sigma^2\right)$$

Now we simulate 50 **new** outcomes based on the same predictor values x:

Y_test = x_train %*% beta + rnorm(n, sd=error_sd)

$$\mathbf{y}_{\text{test}} = \mathbf{X}\cdot\boldsymbol{\beta} + \boldsymbol{\varepsilon}'' \quad \text{with } \varepsilon_i'' \in \mathrm{N}\left(0,\sigma^2\right)$$

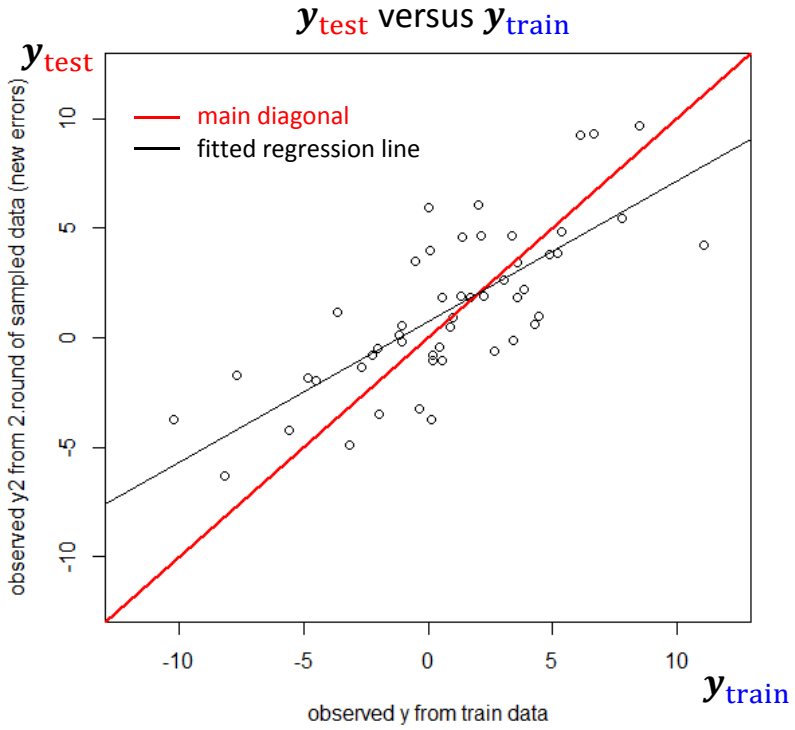
$$\mathbf{y}_{\text{train}} \sim \mathrm{N}\left(\mathbf{X}\cdot\boldsymbol{\beta},\ \sigma^2\mathbf{I}\right)$$

$$\mathbf{y}_{\text{test}} \sim \mathrm{N}\left(\mathbf{X}\cdot\boldsymbol{\beta},\ \sigma^2\mathbf{I}\right)$$

$$\left|\mathrm{corr}\left(\mathbf{y}_{\text{train}}, \mathbf{y}_{\text{test}}\right)\right| < 1$$

→ y_{test} vs y_{train} will show

"shrinkage to the mean"



How to understand why naïve predictions are biased:
Regression to the mean as step on the way

train LS fitted values on train:

$$\hat{\mathbf{y}}_{\text{fitted train}} = \mathbf{X}\cdot\hat{\boldsymbol{\beta}}_{\text{train}} = \mathbf{X}\cdot\boldsymbol{\beta}+\boldsymbol{\delta}',\ \delta'\in V,\ \mathrm{E}(\boldsymbol{\delta}')=0,\ \boldsymbol{\delta}'\perp\mathbf{X}\cdot\boldsymbol{\beta}$$

see Copas'97

train LS prediction on test:

$$\hat{\mathbf{y}}_{\text{pred on test}} = \mathbf{X}\cdot\hat{\boldsymbol{\beta}}_{\text{train}} = \hat{\mathbf{y}}_{\text{train}} = \mathbf{X}\cdot\boldsymbol{\beta}+\boldsymbol{\delta}'$$

$$\hat{\mathbf{y}}_{\text{pred on test}} \sim V \text{ and } \hat{\mathbf{y}}_{\text{fitted test}} \sim V,$$

$$|\mathrm{corr}\left(\hat{\mathbf{y}}_{\text{pred on test}}, \hat{\mathbf{y}}_{\text{fitted test}}\right)<1$$

test LS fitted values on test:

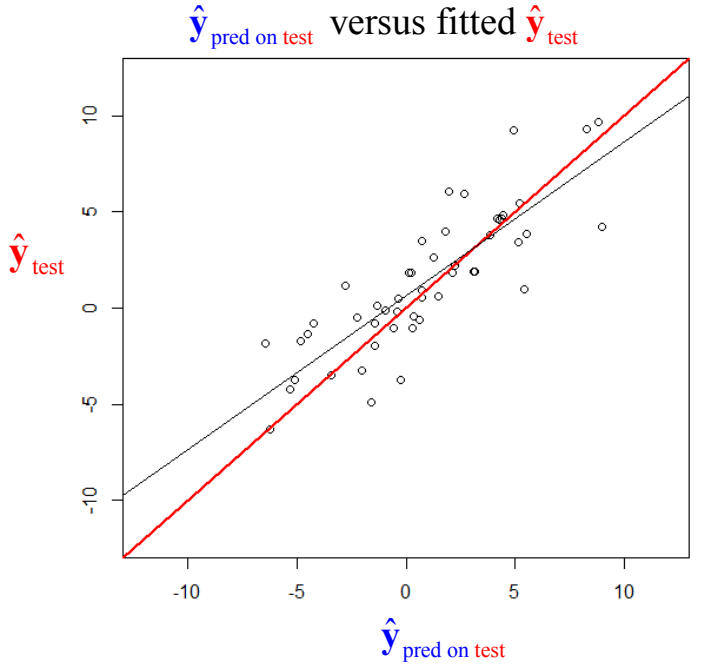
$$\hat{\mathbf{y}}_{\text{fitted test}} = \mathbf{X}\cdot\hat{\boldsymbol{\beta}}_{\text{test}} = \mathbf{X}\cdot\boldsymbol{\beta}+\boldsymbol{\delta}'',\quad \text{with } \delta''\in V,\ \mathrm{E}(\boldsymbol{\delta}'')=0,\ \boldsymbol{\delta}''\perp\mathbf{X}\cdot\boldsymbol{\beta}$$

We are again in a situation where regression to the mean is expected.

$$\mathrm{E}\left(\hat{\mathbf{y}}_{\text{pred on test}} \mid \hat{\mathbf{y}}_{\text{fitted test}}\right) = \rho\cdot\hat{\mathbf{y}}_{\text{pred on test}} + (1-\rho)\cdot\mu\ ,$$

$$\text{with } \mu=\beta^T\mathbf{X} \text{ and } \rho = \frac{\beta^T\cdot\mathrm{Cov}\left(\hat{\mathbf{y}}_{\text{pred on test}}, \hat{\mathbf{y}}_{\text{fitted test}}\right)\cdot\beta}{\beta^T\cdot\mathrm{Cov}\left(\hat{\mathbf{y}}_{\text{pred on test}}, \hat{\mathbf{y}}_{\text{fitted test}}\right)\cdot\beta + \left(\frac{p+1}{n}\right)\cdot\sigma^2}$$

For derivation of ρ see Copas'97
p: # predictors
n: # observations



How to understand why naïve predictions are biased:
using regression to the mean as step on the way

Now let y_{test} be the *actual* observed value of the test observation with covariates $\boldsymbol{X}$.

We know that we get unbiased fitted values with coefficients estimated on test data:

$$\mathrm{E}\left(y_{test} \mid \hat{\mathbf{y}}_{\text{fitted test}} \right) = \hat{\mathbf{y}}_{\text{fitted test}}$$

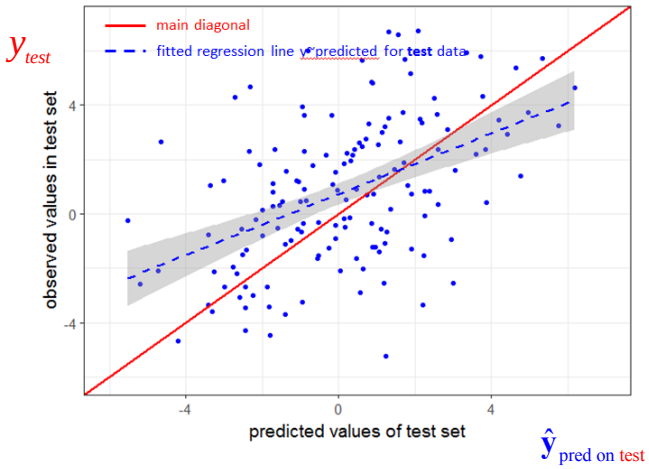
However:

$$\mathrm{E}\left(y_{test} \mid \hat{\mathbf{y}}_{\text{pred on test}} \right) = \mathrm{E}\left(\beta^T \mathbf{X} + \varepsilon'' \mid \hat{\mathbf{y}}_{\text{pred on test}} \right) = \mathrm{E}\left(\hat{\mathbf{y}}_{\text{fitted test}} + \delta'' + \varepsilon'' \mid \hat{\mathbf{y}}_{\text{pred on test}} \right)$$

$$= \mathrm{E}\left(\hat{\mathbf{y}}_{\text{fitted test}} \mid \hat{\mathbf{y}}_{\text{pred on test}} \right) + 0 + 0 \overset{\text{(last slide)}}{=} \rho \cdot \hat{\mathbf{y}}_{\text{pred on test}} + (1-\rho) \cdot \mu \,,$$

$$\text{with } \mu = \beta^T \mathbf{X} \text{ and } \rho = \frac{\beta^T \cdot \mathrm{Cov}\left(\hat{\mathbf{y}}_{\text{pred on test}}, \hat{\mathbf{y}}_{\text{fitted test}} \right) \cdot \beta}{\beta^T \cdot \mathrm{Cov}\left(\hat{\mathbf{y}}_{\text{pred on test}}, \hat{\mathbf{y}}_{\text{fitted test}} \right) \cdot \beta + \left(\frac{p+1}{n} \right) \cdot \sigma^2}$$

$$\Rightarrow \mathrm{E}\left(y_{test} \mid \hat{\mathbf{y}}_{\text{pred on test}} \right) \neq \hat{\mathbf{y}}_{\text{pred on test}}$$

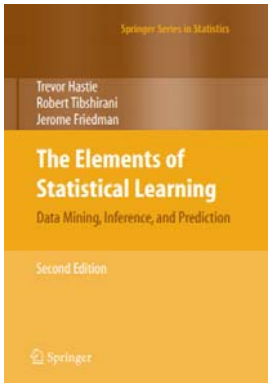


Copas: "Thus, when predicting a new case, the distribution of the actual Y is not centered on the (train) least squares predictor, but regresses towards the mean"

Shrinkage leads to biased estimates on training data.
But: A biased model can have a better prediction performance

On training data estimated prediction model:

$$\hat{f} : X \rightarrow \text{prediction } \hat{f}(X)$$

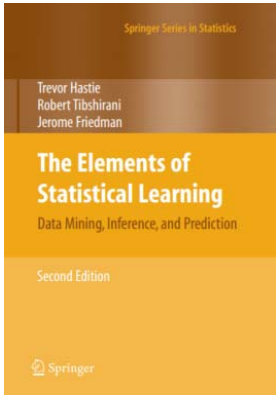


The integrated squared prediction error (EPE) combines both bias and variance in a single summary:

$$\begin{aligned} \text{EPE}(\hat{f}_\lambda) &= \text{E}(Y - \hat{f}_\lambda(X))^2 \\ &= \text{Var}(Y) + \text{E}\left[\text{Bias}^2(\hat{f}_\lambda(X)) + \text{Var}(\hat{f}_\lambda(X))\right] \\ &= \sigma^2 + \text{MSE}(\hat{f}_\lambda). \end{aligned} \tag{5.25}$$

Note that this is averaged both over the training sample (giving rise to $\hat{f}_\lambda$), and the values of the (independently chosen) prediction points (X, Y). EPE is a natural quantity of interest, and does create a tradeoff between bias and variance.

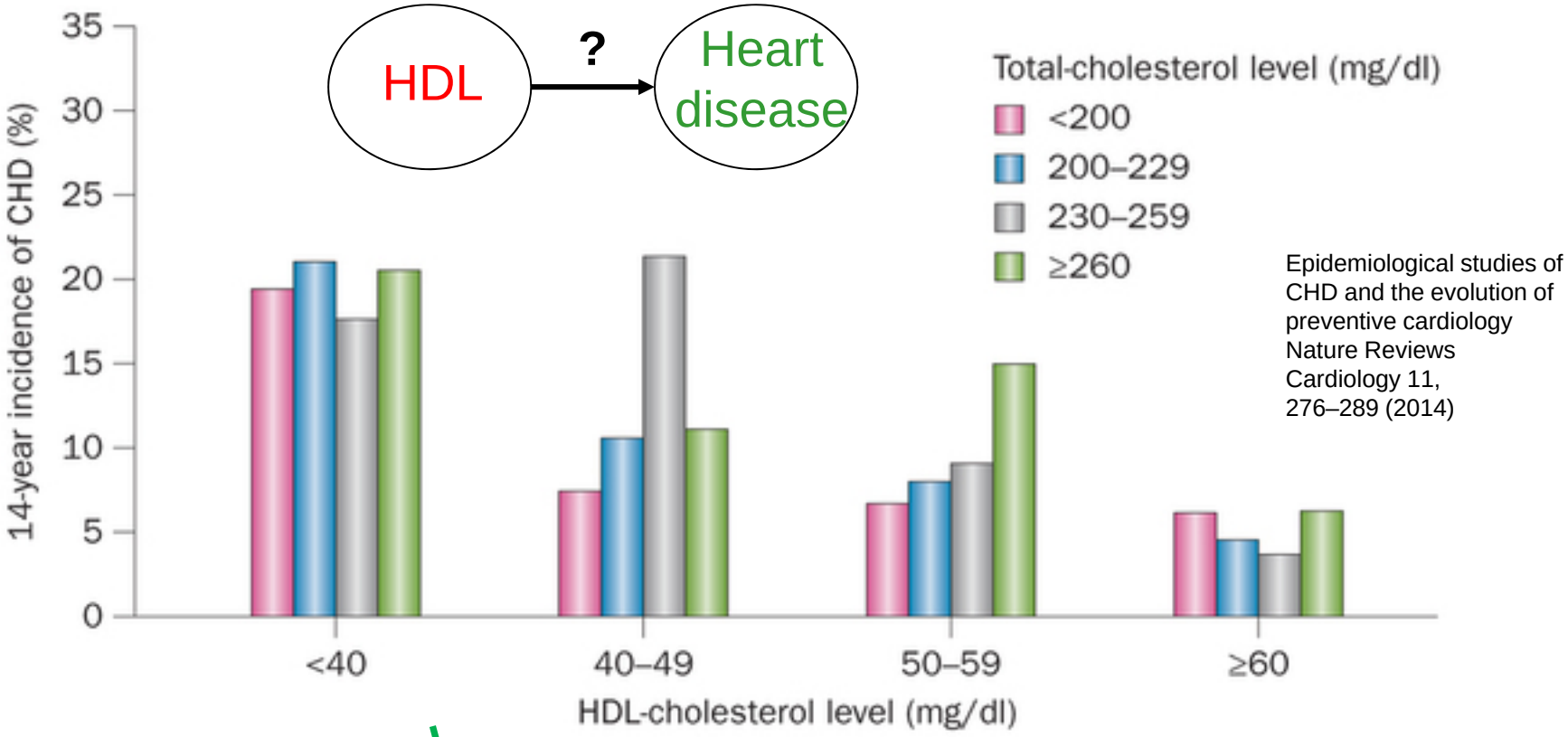
Shrinkage often leads to better prediction performance



Estimated coefficients and test error results, for different subset and shrinkage methods applied to the prostate data. The blank entries correspond to variables omitted.

Term	LS	Best Subset	Ridge	Lasso	PCR	PLS
Intercept	2.465	2.477	2.452	2.468	2.497	2.452
lcavol	0.680	0.740	0.420	0.533	0.543	0.419
lweight	0.263	0.316	0.238	0.169	0.289	0.344
age	−0.141		−0.046		−0.152	−0.026
lbph	0.210		0.162	0.002	0.214	0.220
svi	0.305		0.227	0.094	0.315	0.243
lcp	−0.288		0.000		−0.051	0.079
gleason	−0.021		0.040		0.232	0.011
pgg45	0.267		0.133		−0.056	0.084
Test Error	0.521	0.492	0.492	0.479	0.449	0.528
Std Error	0.179	0.143	0.165	0.164	0.105	0.152

A good predictive model needs not to be a good causal model



Epidemiological studies of CHD and the evolution of preventive cardiology Nature Reviews Cardiology 11, 276–289 (2014)

HDL gives a strong negative association with heart disease in cross-sectional studies and is the strongest predictor of future events in prospective studies.

Prediction worked

Roche tested the effect of drug "dalcetrapib" in phase III on 15'000 patients which proved to boost HDL ("good cholesterol") but failed to prevent heart diseases. Roche stopped the failed trial on May 2012 and immediately lost $5billion of its market capitalization.

Planned intervention failed

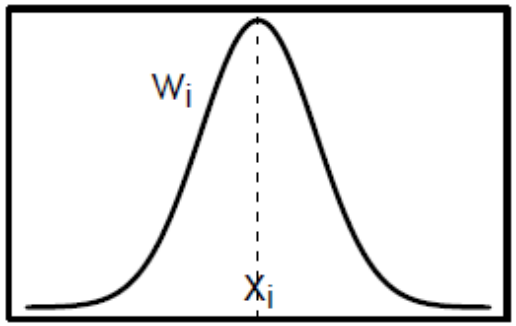
Thank You

Appendix: Errors in explanatory variables

In regression we assume that the explanatory variables are fixed and error-free.

However, in many situations we should assume that the error-free x cannot be observed and instead we observe an error-prone value w.

Classical ME model

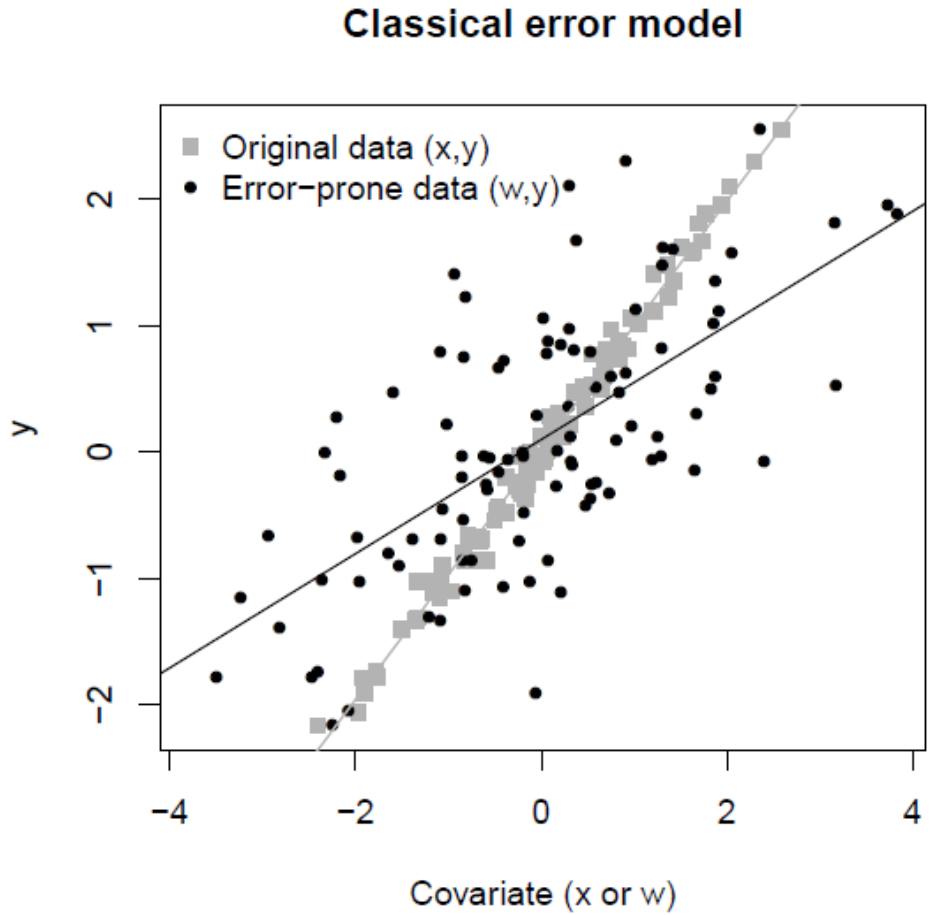


$$W_i = x_i + U_i$$

$$U_i \sim N(0, \sigma_u^2)$$

Appendix: Errors in explanatory variables ctd.

The classical error model (the observed value w_i varies around the true value x_i) is quite common in case of observational data. When fitting a linear regression model based on the error-prone w_i the slope will be attenuated (to small).



Appendix: Errors in explanatory variables ctd.

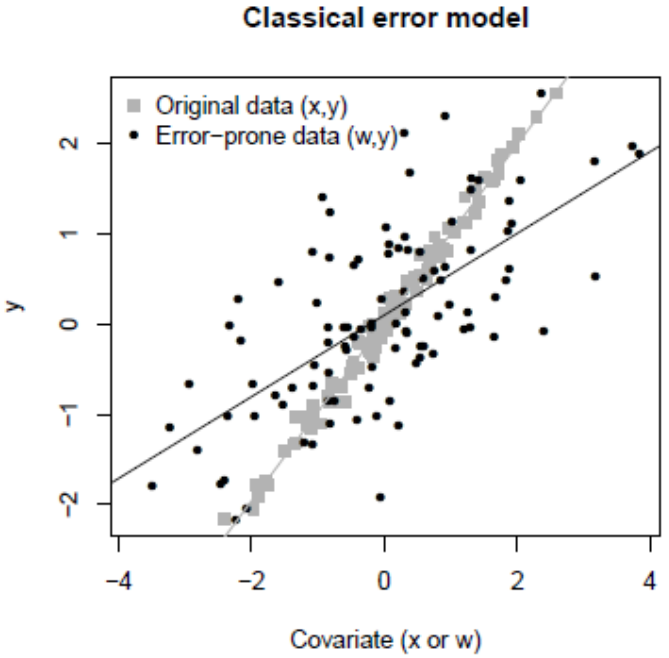
Regression to the mean appears in the **test-retest situation** since both test observations are error-prone, hence we are in the situation of a classical ME model.

We show now that the slope is underestimated in a naïve model in case of classical error structure of the corresponding predictor:

true model: $y = \beta_0 + \beta_x \cdot x + \varepsilon$

error model: $w = x + u \quad , \quad u \perp y, \varepsilon$

naive model: $y = \beta_0^* + \beta_x^* \cdot w + \varepsilon$



estimate with true predictor: $\hat{\beta}_x = \dfrac{\mathrm{cov}(x, y)}{\mathrm{var}(x)}$

naive estimate: $\hat{\beta}_x^* = \dfrac{\mathrm{cov}(w, y)}{\mathrm{var(w)}} = \dfrac{\mathrm{cov}(x+u, y)}{\mathrm{var(x+u)}} = \dfrac{\mathrm{cov}(x, y)}{\mathrm{var(x)} + \mathrm{var(u)}} < \hat{\beta}_x$

Literature

G Shmueli, *To Explain or to Predict? Statistical Science (*2010), Vol. 25, No. 3, 289–310

JB Copas, *Using regression models for prediction: shrinkage and regression to the mean,* Statistical Methods in Medical Research (1997); 6: 167±183

JB Copas, *Regression, Prediction and Shrinkage*, Journal of the Royal Statistical Society. Series B (Methodological), Vol. 45, No. 3(1983), pp. 311-354

BV Calster et al., *A calibration hierarchy for risk models was defined: from utopia to empirical data,* Journal of Clinical Epidemiology 74 (2016); 167e176

JC van Houwelingen, *Shrinkage and penalized likelihood as methods to improve predictive accuracy,* Statistica Neerlandica (2001) Vol. 55, nr. 1, pp. 17±34