Cross-Validation and Dimension Reduction Methods

TrungTin Nguyen

STATIFY team, Inria centre at the University Grenoble Alpes, France

Statistical Analysis and Document Mining

Complementary Course, Master of Applied Mathematics in Grenoble

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Previous episode: Training error versus test error

- The training error can be easily calculated by applying the statistical learning method to the observations used in its training.

Previous episode: Training error versus test error

- The training error can be easily calculated by applying the statistical learning method to the observations used in its training.
- In contrast, the test error is the average error that results from using a statistical learning method to predict the response on a new observation, one that was not used in training the method.

Previous episode: Training error versus test error

- The training error can be easily calculated by applying the statistical learning method to the observations used in its training.
- In contrast, the test error is the average error that results from using a statistical learning method to predict the response on a new observation, one that was not used in training the method.
- $\neq$ The training error rate often is quite different from the test error rate, and in particular the former can **dramatically underestimate** the latter.

Previous episode: Training error versus test error

- The training error can be easily calculated by applying the statistical learning method to the observations used in its training.
- In contrast, the test error is the average error that results from using a statistical learning method to predict the response on a new observation, one that was not used in training the method.

≠ The training error rate often is quite different from the test error rate, and in particular the former can **dramatically underestimate** the latter.

❓ **Not clear enough for you?**

Previous episode: Training error versus test error

- The training error can be easily calculated by applying the statistical learning method to the observations used in its training.
- In contrast, the test error is the average error that results from using a statistical learning method to predict the response on a new observation, one that was not used in training the method.

≠ The training error rate often is quite different from the test error rate, and in particular the former can **dramatically underestimate** the latter.

Not clear enough for you?

In the next slide we will discuss this in more detail?

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Test error in linear regression

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n \in [N]}$, $[N] \equiv 1, \ldots, N$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.

Test error in linear regression

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n\in[N]}$, $[N] \equiv 1, \ldots, N$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.
- Given any error term ϵ, multiple linear regression function takes the form

$$Y = \beta_0 + \sum_{p=1}^{P} \beta_p X_p + \epsilon \equiv r(\mathbf{X}) + \epsilon, \quad \boldsymbol{\beta} \equiv (\beta_0, \beta_1, \ldots, \beta_P).$$

Test error in linear regression

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n\in[N]}$, $[N] \equiv 1, \ldots, N$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.
- Given any error term ϵ, multiple linear regression function takes the form

$$Y = \beta_0 + \sum_{p=1}^{P} \beta_p X_p + \epsilon \equiv r(\mathbf{X}) + \epsilon, \quad \boldsymbol{\beta} \equiv (\beta_0, \beta_1, \ldots, \beta_P).$$

- **Training error:** using ordinary least squares (OLS), we obtain coefficient estimates $\widehat{\boldsymbol{\beta}}$ and $\widehat{r}_{\mathcal{D}}(\mathbf{x}_n) \equiv \widehat{\beta}_0 + \sum_{p=1}^{P} \widehat{\beta}_p x_{np}$ such that $\forall n \in [N]$,

$$y_n \approx \widehat{r}_{\mathcal{D}}(\mathbf{x}_n), \text{ or equivalent, } \mathcal{L}(\widehat{r}_{\mathcal{D}}, \mathcal{D}) \equiv \frac{1}{N} \sum_{n=1}^{N} (y_n - \widehat{r}_{\mathcal{D}}(\mathbf{x}_n))^2 \approx 0. \quad (1)$$

Test error in linear regression

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n\in[N]}$, $[N] \equiv 1, \ldots, N$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.
- Given any error term ϵ, multiple linear regression function takes the form

$$Y = \beta_0 + \sum_{p=1}^{P} \beta_p X_p + \epsilon \equiv r(\mathbf{X}) + \epsilon, \quad \boldsymbol{\beta} \equiv (\beta_0, \beta_1, \ldots, \beta_P).$$

- **Training error:** using ordinary least squares (OLS), we obtain coefficient estimates $\widehat{\boldsymbol{\beta}}$ and $\widehat{r}_{\mathcal{D}}(\mathbf{x}_n) \equiv \widehat{\beta}_0 + \sum_{p=1}^{P} \widehat{\beta}_p x_{np}$ such that $\forall n \in [N]$,

$$y_n \approx \widehat{r}_{\mathcal{D}}(\mathbf{x}_n), \text{ or equivalent, } \mathcal{L}(\widehat{r}_{\mathcal{D}}, \mathcal{D}) \equiv \frac{1}{N} \sum_{n=1}^{N} (y_n - \widehat{r}_{\mathcal{D}}(\mathbf{x}_n))^2 \approx 0. \quad (1)$$

- **Test (generalization) error:** for any **new sample** $(\mathbf{x}^*, y^*)$, how can we guarantee

$$y^* \approx \widehat{r}_{\mathcal{D}}(\mathbf{x}^*), \text{ or equivalent, } \mathcal{L}(\widehat{r}_{\mathcal{D}}) \equiv \mathbb{E}_{\mathbf{X},Y}\left[(Y - \widehat{r}_{\mathcal{D}}(\mathbf{X}))^2\right] \approx 0? \quad (2)$$

Recall from CM3, in general, it holds that $\mathcal{L}(\widehat{r}_{\mathcal{D}}, \mathcal{D}) \leq \mathcal{L}(\widehat{r}_{\mathcal{D}})$. (3)

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Previous episode: the general model selection paradigm

- **Model selection problem:** let $(S_m)_{m\in\mathcal{M}}$ be a family of models. For every $m \in \mathcal{M}$, let $\widehat{s}_m(\mathcal{D}_N)$ be a minimum contrast estimator, *e.g.,* least-squares contrast in LS, over S_m. **Our goal is to choose the best data-driven model $\widehat{m} \equiv \widehat{m}(\mathcal{D}_N) \in \mathcal{M}$ from data.**

[1] Mallows, C. L. (1973). "Some Comments on C_P". Technometrics.

[2] Akaike, H. (1974). "A new look at the statistical model identification". IEEE Transactions on Automatic Control.

[3] Schwarz, G. (1978). "Estimating the dimension of a model". The Annals of Statistics.

[4] Stone, M. (1974). "Cross-validatory choice and assessment of statistical predictions". JRSSSB.

[5] Geisser, S. (1975). "The predictive sample reuse method with applications". J. Amer. Statist. Assoc.

[6] Arlot, S., & Celisse, A. (2010). "A survey of cross-validation procedures for model selection". Statist. Surv.

[7] Birgé, L. and Massart, P. (2007). "Minimal penalties for Gaussian model selection". Probability Theory and Related Fields.

[8] **Nguyen, T.**, Nguyen, H. D., Chamroukhi, F., & Forbes, F. (2022). "A non-asymptotic approach for model selection via penalization in high-dimensional mixture of experts models". Electronic Journal of Statistics.

Previous episode: the general model selection paradigm

- **Model selection problem:** let $(S_m)_{m \in \mathcal{M}}$ be a family of models. For every $m \in \mathcal{M}$, let $\widehat{s}_m(\mathcal{D}_N)$ be a minimum contrast estimator, *e.g.,* least-squares contrast in LS, over S_m. **Our goal is to choose the best data-driven model $\widehat{m} \equiv \widehat{m}(\mathcal{D}_N) \in \mathcal{M}$ from data.**
- **Some model selection criteria:**
  1. Asymptotic approach: Mallows's C_p[1],

[1] Mallows, C. L. (1973). "Some Comments on C_P". Technometrics.

[2] Akaike, H. (1974). "A new look at the statistical model identification". IEEE Transactions on Automatic Control.

[3] Schwarz, G. (1978). "Estimating the dimension of a model". The Annals of Statistics.

[4] Stone, M. (1974). "Cross-validatory choice and assessment of statistical predictions". JRSSSB.

[5] Geisser, S. (1975). "The predictive sample reuse method with applications". J. Amer. Statist. Assoc.

[6] Arlot, S., & Celisse, A. (2010). "A survey of cross-validation procedures for model selection". Statist. Surv.

[7] Birgé, L. and Massart, P. (2007). "Minimal penalties for Gaussian model selection". Probability Theory and Related Fields.

[8] **Nguyen, T.**, Nguyen, H. D., Chamroukhi, F., & Forbes, F. (2022). "A non-asymptotic approach for model selection via penalization in high-dimensional mixture of experts models". Electronic Journal of Statistics.

Previous episode: the general model selection paradigm

- **Model selection problem:** let $(S_m)_{m \in \mathcal{M}}$ be a family of models. For every $m \in \mathcal{M}$, let $\widehat{s}_m(\mathcal{D}_N)$ be a minimum contrast estimator, *e.g.,* least-squares contrast in LS, over S_m. **Our goal is to choose the best data-driven model $\widehat{m} \equiv \widehat{m}(\mathcal{D}_N) \in \mathcal{M}$ from data.**
- **Some model selection criteria:**
  1. Asymptotic approach: Mallows's C_p[1], Akaike information criterion[2] (AIC),

[1] Mallows, C. L. (1973). "Some Comments on C_P". Technometrics.

[2] Akaike, H. (1974). "A new look at the statistical model identification". IEEE Transactions on Automatic Control.

[3] Schwarz, G. (1978). "Estimating the dimension of a model". The Annals of Statistics.

[4] Stone, M. (1974). "Cross-validatory choice and assessment of statistical predictions". JRSSSB.

[5] Geisser, S. (1975). "The predictive sample reuse method with applications". J. Amer. Statist. Assoc.

[6] Arlot, S., & Celisse, A. (2010). "A survey of cross-validation procedures for model selection". Statist. Surv.

[7] Birgé, L. and Massart, P. (2007). "Minimal penalties for Gaussian model selection". Probability Theory and Related Fields.

[8] **Nguyen, T.**, Nguyen, H. D., Chamroukhi, F., & Forbes, F. (2022). "A non-asymptotic approach for model selection via penalization in high-dimensional mixture of experts models". Electronic Journal of Statistics.

Previous episode: the general model selection paradigm

- **Model selection problem:** let $(S_m)_{m\in\mathcal{M}}$ be a family of models. For every $m \in \mathcal{M}$, let $\widehat{s}_m(\mathcal{D}_N)$ be a minimum contrast estimator, *e.g.,* least-squares contrast in LS, over S_m. **Our goal is to choose the best data-driven model $\widehat{m} \equiv \widehat{m}(\mathcal{D}_N) \in \mathcal{M}$ from data.**
- **Some model selection criteria:**
  1. Asymptotic approach: Mallows's C_p[1], Akaike information criterion[2] (AIC), Bayesian information criterion[3] (BIC): no finite sample guarantees.

[1] Mallows, C. L. (1973). "Some Comments on C_P". Technometrics.

[2] Akaike, H. (1974). "A new look at the statistical model identification". IEEE Transactions on Automatic Control.

[3] Schwarz, G. (1978). "Estimating the dimension of a model". The Annals of Statistics.

[4] Stone, M. (1974). "Cross-validatory choice and assessment of statistical predictions". JRSSSB.

[5] Geisser, S. (1975). "The predictive sample reuse method with applications". J. Amer. Statist. Assoc.

[6] Arlot, S., & Celisse, A. (2010). "A survey of cross-validation procedures for model selection". Statist. Surv.

[7] Birgé, L. and Massart, P. (2007). "Minimal penalties for Gaussian model selection". Probability Theory and Related Fields.

[8] **Nguyen, T.**, Nguyen, H. D., Chamroukhi, F., & Forbes, F. (2022). "A non-asymptotic approach for model selection via penalization in high-dimensional mixture of experts models". Electronic Journal of Statistics.

Previous episode: the general model selection paradigm

- **Model selection problem:** let $(S_m)_{m \in \mathcal{M}}$ be a family of models. For every $m \in \mathcal{M}$, let $\widehat{s}_m(\mathcal{D}_N)$ be a minimum contrast estimator, *e.g.,* least-squares contrast in LS, over S_m. **Our goal is to choose the best data-driven model $\widehat{m} \equiv \widehat{m}(\mathcal{D}_N) \in \mathcal{M}$ from data.**
- **Some model selection criteria:**
  1. Asymptotic approach: Mallows's C_p[1], Akaike information criterion[2] (AIC), Bayesian information criterion[3] (BIC): no finite sample guarantees.
  2. Cross-validation procedures:[4] [5] [6] K-Fold, leave-one-out.

[1] Mallows, C. L. (1973). "Some Comments on C_P". Technometrics.

[2] Akaike, H. (1974). "A new look at the statistical model identification". IEEE Transactions on Automatic Control.

[3] Schwarz, G. (1978). "Estimating the dimension of a model". The Annals of Statistics.

[4] Stone, M. (1974). "Cross-validatory choice and assessment of statistical predictions". JRSSSB.

[5] Geisser, S. (1975). "The predictive sample reuse method with applications". J. Amer. Statist. Assoc.

[6] Arlot, S., & Celisse, A. (2010). "A survey of cross-validation procedures for model selection". Statist. Surv.

[7] Birgé, L. and Massart, P. (2007). "Minimal penalties for Gaussian model selection". Probability Theory and Related Fields.

[8] **Nguyen, T.**, Nguyen, H. D., Chamroukhi, F., & Forbes, F. (2022). "A non-asymptotic approach for model selection via penalization in high-dimensional mixture of experts models". Electronic Journal of Statistics.

Previous episode: the general model selection paradigm

- **Model selection problem:** let $(S_m)_{m \in \mathcal{M}}$ be a family of models. For every $m \in \mathcal{M}$, let $\widehat{s}_m(\mathcal{D}_N)$ be a minimum contrast estimator, *e.g.*, least-squares contrast in LS, over S_m. **Our goal is to choose the best data-driven model $\widehat{m} \equiv \widehat{m}(\mathcal{D}_N) \in \mathcal{M}$ from data.**
- **Some model selection criteria:**
  1. Asymptotic approach: Mallows's C_p[1], Akaike information criterion[2] (AIC), Bayesian information criterion[3] (BIC): no finite sample guarantees.
  2. Cross-validation procedures:[4] [5] [6] K-Fold, leave-one-out.
  3. Non-asymptotic approach: **slope heuristic**[7] [8], which is particularly useful for high-dimensional small data sets, *e.g.*, $N \ll P$.

[1] Mallows, C. L. (1973). "Some Comments on C_P". Technometrics.

[2] Akaike, H. (1974). "A new look at the statistical model identification". IEEE Transactions on Automatic Control.

[3] Schwarz, G. (1978). "Estimating the dimension of a model". The Annals of Statistics.

[4] Stone, M. (1974). "Cross-validatory choice and assessment of statistical predictions". JRSSSB.

[5] Geisser, S. (1975). "The predictive sample reuse method with applications". J. Amer. Statist. Assoc.

[6] Arlot, S., & Celisse, A. (2010). "A survey of cross-validation procedures for model selection". Statist. Surv.

[7] Birgé, L. and Massart, P. (2007). "Minimal penalties for Gaussian model selection". Probability Theory and Related Fields.

[8] **Nguyen, T.**, Nguyen, H. D., Chamroukhi, F., & Forbes, F. (2022). "A non-asymptotic approach for model selection via penalization in high-dimensional mixture of experts models". Electronic Journal of Statistics.

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

K-Fold cross-validation

- **Widely used approach** for estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$.
  1. Idea is to randomly divide the data $\mathcal{D}$ into K parts.

K-Fold cross-validation

- **Widely used approach** for estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$.
  1. Idea is to randomly divide the data $\mathcal{D}$ into K parts.
  2. We leave out part k, fit the model to the other $K-1$ parts (combined).

K-Fold cross-validation

- **Widely used approach** for estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$.
  1. Idea is to randomly divide the data $\mathcal{D}$ into K parts.
  2. We leave out part k, fit the model to the other $K-1$ parts (combined).
  3. We obtain predictions for the left-out kth part.

K-Fold cross-validation

- **Widely used approach** for estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$.
  1. Idea is to randomly divide the data $\mathcal{D}$ into K parts.
  2. We leave out part k, fit the model to the other $K-1$ parts (combined).
  3. We obtain predictions for the left-out kth part.
  4. This is done in turn for each part $k \in [K]$ and then the results are combined.

K-Fold cross-validation

- **Widely used approach** for estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$.
  1. Idea is to randomly divide the data $\mathcal{D}$ into K parts.
  2. We leave out part k, fit the model to the other $K-1$ parts (combined).
  3. We obtain predictions for the left-out kth part.
  4. This is done in turn for each part $k \in [K]$ and then the results are combined.
- Estimates can be used to select best model, and to give an idea of the test error of the final chosen model.

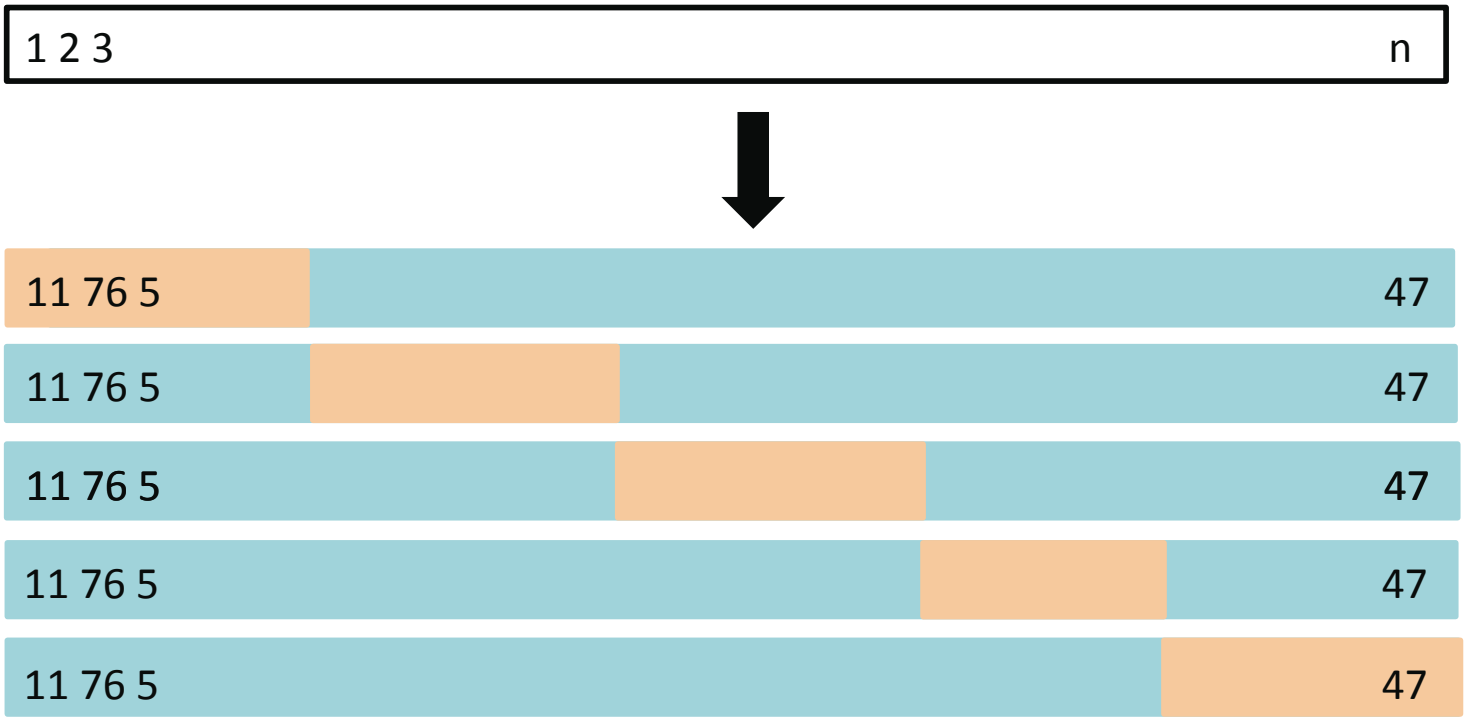


Figure 1: A schematic display of 5-fold CV. A set of N observations is randomly split into five non-overlapping groups. Each of these fifths acts as a validation set (shown in beige), and the remainder as a training set (shown in blue). The test error is estimated by averaging the five resulting estimates [James et al., 2021, Figure 5.5].

K-Fold cross-validation for linear regression in detail

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.

K-Fold cross-validation for linear regression in detail

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.

K-Fold cross-validation for linear regression in detail

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k}\left(y_n - \widehat{r}^{(-k)}(\mathbf{x}_n)\right)^2\right]}_{\text{Estimate test error for each fold}} \equiv \mathrm{CV}(\widehat{r}, K, \mathbf{m}).$$

K-Fold cross-validation for linear regression in detail

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k}\left(y_n - \widehat{r}^{(-k)}(\mathbf{x}_n)\right)^2\right]}_{\text{Estimate test error for each fold}} \equiv \mathrm{CV}(\widehat{r}, K, \mathbf{m}).$$

4. Best data-driven model: $\widehat{\mathbf{m}} \equiv \mathrm{argmin}_{\mathbf{m}\in\mathcal{M}} \mathrm{CV}(\widehat{r}, K, \mathbf{m})$.

K-Fold cross-validation for linear regression in detail

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k}\left(y_n - \widehat{r}^{(-k)}(\mathbf{x}_n)\right)^2\right]}_{\text{Estimate test error for each fold}} \equiv \mathrm{CV}(\widehat{r}, K, \mathbf{m}).$$

4. Best data-driven model: $\widehat{\mathbf{m}} \equiv \mathrm{argmin}_{\mathbf{m}\in\mathcal{M}} \mathrm{CV}(\widehat{r}, K, \mathbf{m})$.
   K = ?

K-Fold cross-validation for linear regression in detail

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k}\left(y_n - \widehat{r}^{(-k)}(\mathbf{x}_n)\right)^2\right]}_{\text{Estimate test error for each fold}} \equiv \mathrm{CV}(\widehat{r}, K, \mathbf{m}).$$

4. Best data-driven model: $\widehat{\mathbf{m}} \equiv \mathrm{argmin}_{\mathbf{m}\in\mathcal{M}} \mathrm{CV}(\widehat{r}, K, \mathbf{m})$.
   K = ? Setting $K = N$ yields N-fold or leave-one out cross-validation (LOOCV, **high variance**).

K-Fold cross-validation for linear regression in detail

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k}\left(y_n - \widehat{r}^{(-k)}(\mathbf{x}_n)\right)^2\right]}_{\text{Estimate test error for each fold}} \equiv \mathrm{CV}(\widehat{r}, K, \mathbf{m}).$$

4. Best data-driven model: $\widehat{\mathbf{m}} \equiv \mathrm{argmin}_{\mathbf{m}\in\mathcal{M}} \mathrm{CV}(\widehat{r}, K, \mathbf{m})$.
   K = ? Setting $K = N$ yields N-fold or leave-one out cross-validation (LOOCV, **high variance**). A common better choice $K = 5$ or 10.

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Test error in binary classification: more details in CM5

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n\in[N]}$, $y_{[N]} \in [2] \equiv \mathcal{C}$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.

Test error in binary classification: more details in CM5

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n \in [N]}$, $y_{[N]} \in [2] \equiv \mathcal{C}$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.
- How to model relationship between $p_c(\mathbf{X}) = \mathbb{P}(Y = c|\mathbf{X})$, $c \in \mathcal{C}$, and $\mathbf{X}$?

Test error in binary classification: more details in CM5

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n\in[N]}$, $y_{[N]} \in [2] \equiv \mathcal{C}$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.
- How to model relationship between $p_c(\mathbf{X}) = \mathbb{P}(Y = c|\mathbf{X})$, $c \in \mathcal{C}$, and $\mathbf{X}$?
- The multiple logistic regression (MLR) takes the form $p_2(\mathbf{X}) = 1 - p_1(\mathbf{X})$, where

$$\log\left(\frac{p_1(\mathbf{X})}{1-p_1(\mathbf{X})}\right) = \beta_0 + \sum_{p=1}^{P}\beta_p X_p, \text{ or } p_1(\mathbf{X}) = \frac{\exp(\beta_0 + \sum_{p=1}^{P}\beta_p X_p)}{1+\exp(\beta_0 + \sum_{p=1}^{P}\beta_p X_p)}.$$

Test error in binary classification: more details in CM5

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n\in[N]}$, $y_{[N]} \in [2] \equiv \mathcal{C}$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.
- How to model relationship between $p_c(\mathbf{X}) = \mathbb{P}(Y = c|\mathbf{X})$, $c \in \mathcal{C}$, and $\mathbf{X}$?
- The multiple logistic regression (MLR) takes the form $p_2(\mathbf{X}) = 1 - p_1(\mathbf{X})$, where

$$\log\left(\frac{p_1(\mathbf{X})}{1-p_1(\mathbf{X})}\right) = \beta_0 + \sum_{p=1}^{P}\beta_p X_p, \text{ or } p_1(\mathbf{X}) = \frac{\exp(\beta_0 + \sum_{p=1}^{P}\beta_p X_p)}{1+\exp(\beta_0 + \sum_{p=1}^{P}\beta_p X_p)}.$$

- **Training error:** using non-linear LS or maximum likelihood estimation (MLE), we obtain $\widehat{\beta}$ and $\widehat{r}_{\mathcal{D}}(\mathbf{x}_n) = \text{argmax}_{c\in\mathcal{C}}\, p_c(\mathbf{x}_n)$ such that $\forall n \in [N]$,

$$y_n \approx \widehat{r}_{\mathcal{D}}(\mathbf{x}_n), \text{ or equivalent, } \mathcal{L}(\widehat{r}_{\mathcal{D}}, \mathcal{D}) \equiv \frac{1}{N}\sum_{n=1}^{N}\mathbb{1}\left[y_n \neq \widehat{r}_{\mathcal{D}}(\mathbf{x}_n)\right] \approx 0. \quad (4)$$

Test error in binary classification: more details in CM5

- We are given a **training dataset** $\mathcal{D} \equiv \{(\mathbf{x}_n, y_n)\}_{n\in[N]}$, $y_{[N]} \in [2] \equiv \mathcal{C}$, i.i.d. sampled from **the true (but unknown) joint PDF** of $(\mathbf{X}, Y)$.
- How to model relationship between $p_c(\mathbf{X}) = \mathbb{P}(Y = c|\mathbf{X})$, $c \in \mathcal{C}$, and $\mathbf{X}$?
- The multiple logistic regression (MLR) takes the form $p_2(\mathbf{X}) = 1 - p_1(\mathbf{X})$, where

$$\log\left(\frac{p_1(\mathbf{X})}{1-p_1(\mathbf{X})}\right) = \beta_0 + \sum_{p=1}^{P}\beta_p X_p, \text{ or } p_1(\mathbf{X}) = \frac{\exp(\beta_0 + \sum_{p=1}^{P}\beta_p X_p)}{1+\exp(\beta_0 + \sum_{p=1}^{P}\beta_p X_p)}.$$

- **Training error:** using non-linear LS or maximum likelihood estimation (MLE), we obtain $\widehat{\beta}$ and $\widehat{r}_{\mathcal{D}}(\mathbf{x}_n) = \text{argmax}_{c\in\mathcal{C}}\, p_c(\mathbf{x}_n)$ such that $\forall n \in [N]$,

$$y_n \approx \widehat{r}_{\mathcal{D}}(\mathbf{x}_n), \text{ or equivalent, } \mathcal{L}(\widehat{r}_{\mathcal{D}}, \mathcal{D}) \equiv \frac{1}{N}\sum_{n=1}^{N}\mathbb{1}\left[y_n \neq \widehat{r}_{\mathcal{D}}(\mathbf{x}_n)\right] \approx 0. \quad (4)$$

- **Test (generalization) error:** for any **new sample** $(\mathbf{x}^*, y^*)$, how can we guarantee

$$y^* \approx \widehat{r}_{\mathcal{D}}(\mathbf{x}^*), \text{ or equivalent, } \mathcal{L}(\widehat{r}_{\mathcal{D}}) \equiv \mathbb{E}_{\mathbf{X},Y}\left[\mathbb{1}\left(Y \neq \widehat{r}_{\mathcal{D}}(\mathbf{X})\right)\right] \approx 0? \quad (5)$$

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Cross-Validation for classification problems

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.

Cross-Validation for classification problems

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.

Cross-Validation for classification problems

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k} \mathbb{1}\left(y_n \neq \widehat{r}^{(-k)}(\mathbf{x}_n)\right)\right]}_{\text{Estimate test error for each fold}} \equiv \mathrm{CV}(\widehat{r}, K, \mathbf{m}).$$

Cross-Validation for classification problems

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k} \mathbb{1}\left(y_n \neq \widehat{r}^{(-k)}(\mathbf{x}_n)\right)\right]}_{\text{Estimate test error for each fold}} \equiv \mathrm{CV}(\widehat{r}, K, \mathbf{m}).$$

4. Best data-driven model: $\widehat{\mathbf{m}} \equiv \mathrm{argmin}_{\mathbf{m}\in\mathcal{M}} \mathrm{CV}(\widehat{r}, K, \mathbf{m})$.

Cross-Validation for classification problems

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k} \mathbb{1}\left(y_n \neq \widehat{r}^{(-k)}(\mathbf{x}_n)\right)\right]}_{\text{Estimate test error for each fold}} \equiv \mathrm{CV}(\widehat{r}, K, \mathbf{m}).$$

4. Best data-driven model: $\widehat{\mathbf{m}} \equiv \mathrm{argmin}_{\mathbf{m}\in\mathcal{M}} \mathrm{CV}(\widehat{r}, K, \mathbf{m})$.
   K = ?

Cross-Validation for classification problems

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k} \mathbb{1}\left(y_n \neq \widehat{r}^{(-k)}(\mathbf{x}_n)\right)\right]}_{\text{Estimate test error for each fold}} \equiv \text{CV}(\widehat{r}, K, \mathbf{m}).$$

4. Best data-driven model: $\widehat{\mathbf{m}} \equiv \text{argmin}_{\mathbf{m}\in\mathcal{M}} \text{CV}(\widehat{r}, K, \mathbf{m})$.
   K = ? Setting $K = N$ yields N-fold or leave-one out cross-validation (LOOCV, **high variance**).

Cross-Validation for classification problems

1. Split the training dataset randomly into K folds so that we have $\mathcal{D}_1 \cup \cdots \cup \mathcal{D}_K = \mathcal{D}$, where $\mathcal{D}_k$ denotes the indices of the observations in part k. There are N_k observations in part k: if N is a multiple of K, then $N_k = N/K$.
2. For each $k \in [K]$, fit a model $\widehat{r}^{(-k)}(\mathbf{x}, \mathbf{m})$, indexed by a tuning parameter (or a model) $\mathbf{m} \in \mathcal{M}$, on all samples from the training set except those in the kth fold.
3. Estimating test error $\mathcal{L}(\widehat{r}_{\mathcal{D}})$ via averaging the final resulting MSE estimates

$$\mathcal{L}(\widehat{r}_{\mathcal{D}}) \approx \underbrace{\frac{1}{K}\sum_{k=1}^{K}}_{\text{average over } K \text{ folds}} \underbrace{\left[\frac{1}{N_k}\sum_{n\in\mathcal{D}_k} \mathbb{1}\left(y_n \neq \widehat{r}^{(-k)}(\mathbf{x}_n)\right)\right]}_{\text{Estimate test error for each fold}} \equiv \text{CV}(\widehat{r}, K, \mathbf{m}).$$

4. Best data-driven model: $\widehat{\mathbf{m}} \equiv \text{argmin}_{\mathbf{m}\in\mathcal{M}} \text{CV}(\widehat{r}, K, \mathbf{m})$.
   K = ? Setting $K = N$ yields N-fold or leave-one out cross-validation (LOOCV, **high variance**). A common better choice $K = 5$ or 10.

Cross-Validation: right and wrong

Consider a simple classifier applied to some two-class data with a **large number of** 5000 **predictors and a small number of** 50 **samples**. This may arise in genetics (TP2-TP3) or proteomic applications, for example.

Cross-Validation: right and wrong

Consider a simple classifier applied to some two-class data with a **large number of** 5000 **predictors and a small number of** 50 **samples**. This may arise in genetics (TP2-TP3) or proteomic applications, for example. How do we estimate the test set performance of this classifier?

Cross-Validation: right and wrong

Consider a simple classifier applied to some two-class data with a **large number of** 5000 **predictors and a small number of** 50 **samples**. This may arise in genetics (TP2-TP3) or proteomic applications, for example.
How do we estimate the test set performance of this classifier?
The **right way** to do cross-validation:

Cross-Validation: right and wrong

Consider a simple classifier applied to some two-class data with a **large number of** 5000 **predictors and a small number of** 50 **samples**. This may arise in genetics (TP2-TP3) or proteomic applications, for example.
How do we estimate the test set performance of this classifier?
The **right way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier,*e.g.*, MLR.
3. Use cross-validation to estimate the unknown tuning parameters $\mathbf{m}$ and to estimate the prediction error, *e.g.*, $\mathrm{CV}(\hat{r}, K, \hat{\mathbf{m}})$, of the final model.

Cross-Validation: right and wrong

Consider a simple classifier applied to some two-class data with a **large number of** 5000 **predictors and a small number of** 50 **samples**. This may arise in genetics (TP2-TP3) or proteomic applications, for example.
How do we estimate the test set performance of this classifier?
The **right way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier,*e.g.*, MLR.
3. Use cross-validation to estimate the unknown tuning parameters $\mathbf{m}$ and to estimate the prediction error, *e.g.*, $\mathrm{CV}(\hat{r}, K, \hat{\mathbf{m}})$, of the final model.

Wait! Is this a correct application of cross-validation?

Cross-Validation: right and wrong

Consider a simple classifier applied to some two-class data with a **large number of** 5000 **predictors and a small number of** 50 **samples**. This may arise in genetics (TP2-TP3) or proteomic applications, for example.
How do we estimate the test set performance of this classifier?
The **right way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier,*e.g.*, MLR.
3. Use cross-validation to estimate the unknown tuning parameters $\mathbf{m}$ and to estimate the prediction error, *e.g.*, $\mathrm{CV}(\hat{r}, K, \hat{\mathbf{m}})$, of the final model.

Wait! Is this a correct application of cross-validation?
Can we apply cross-validation in step 2, forgetting about step 1?

Cross-Validation: right and wrong

Consider a simple classifier applied to some two-class data with a **large number of** 5000 **predictors and a small number of** 50 **samples**. This may arise in genetics (TP2-TP3) or proteomic applications, for example.
How do we estimate the test set performance of this classifier?
The **right way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier,*e.g.*, MLR.
3. Use cross-validation to estimate the unknown tuning parameters $\mathbf{m}$ and to estimate the prediction error, *e.g.*, $\mathrm{CV}(\hat{r}, K, \hat{\mathbf{m}})$, of the final model.

Wait! Is this a correct application of cross-validation?
Can we apply cross-validation in step 2, forgetting about step 1? **NO!**

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Cross-Validation: right and wrong

The **wrong way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier.
3. Use cross-validation to estimate the unknown tuning parameters **m** and to estimate the prediction error of the final model.

Cross-Validation: right and wrong

The **wrong way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier.
3. Use cross-validation to estimate the unknown tuning parameters **m** and to estimate the prediction error of the final model.

? BUT why:

Cross-Validation: right and wrong

The **wrong way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier.
3. Use cross-validation to estimate the unknown tuning parameters **m** and to estimate the prediction error of the final model.

? BUT why:

- We can simulate realistic data with the class labels independent of the outcome, so that true test error =50%, but the CV error estimate that ignores Step 1 is almost zero!

Cross-Validation: right and wrong

The **wrong way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier.
3. Use cross-validation to estimate the unknown tuning parameters **m** and to estimate the prediction error of the final model.

? BUT why:

- We can simulate realistic data with the class labels independent of the outcome, so that true test error =50%, but the CV error estimate that ignores Step 1 is almost zero! What has happened?

Cross-Validation: right and wrong

The **wrong way** to do cross-validation:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels.
2. Using just this subset of predictors, build a multivariate classifier.
3. Use cross-validation to estimate the unknown tuning parameters **m** and to estimate the prediction error of the final model.

BUT why:

- We can simulate realistic data with the class labels independent of the outcome, so that true test error =50%, but the CV error estimate that ignores Step 1 is almost zero! What has happened?
- The problem is that the predictors have an unfair advantage, as they were chosen in step (1) on the basis of all of the samples. Leaving samples out after the variables have been selected does not correctly mimic the application of the classifier to a completely independent test set, since these predictors "have already seen" the left out samples.

This error made in many high profile genomics papers [Hastie et al., 2009, Section 7.10.2].

The **right way** to do cross-validation

1. Divide the samples into K cross-validation folds (groups) at random.

The **right way** to do cross-validation

1. Divide the samples into K cross-validation folds (groups) at random.
2. For each fold $k = 1, \ldots, K$:
   a. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels, **using all of the samples except those in fold** k.

The **right way** to do cross-validation

1. Divide the samples into K cross-validation folds (groups) at random.
2. For each fold $k = 1, \ldots, K$:
   a. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels, **using all of the samples except those in fold** k.
   b. Using just this subset of predictors, build a multivariate classifier, **using all of the samples except those in fold** k.

The **right way** to do cross-validation

1. Divide the samples into K cross-validation folds (groups) at random.
2. For each fold $k = 1, \ldots, K$:
   a. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels, **using all of the samples except those in fold** k.
   b. Using just this subset of predictors, build a multivariate classifier, **using all of the samples except those in fold** k.
   c. Use the classifier to predict the class labels **for the samples in fold** k.

The **right way** to do cross-validation

1. Divide the samples into K cross-validation folds (groups) at random.
2. For each fold $k = 1, \ldots, K$:
   a. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels, **using all of the samples except those in fold** k.
   b. Using just this subset of predictors, build a multivariate classifier, **using all of the samples except those in fold** k.
   c. Use the classifier to predict the class labels **for the samples in fold** k.
3. The error estimates from step 2(c) are then **accumulated over all** K **folds**, to produce the cross-validation estimate of prediction error $\mathrm{CV}(\hat{r}, K, \mathbf{m})$ for each model $\mathbf{m} \in \mathcal{M}$.

The **right way** to do cross-validation

1. Divide the samples into K cross-validation folds (groups) at random.
2. For each fold $k = 1, \ldots, K$:
   a. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels, **using all of the samples except those in fold** k.
   b. Using just this subset of predictors, build a multivariate classifier, **using all of the samples except those in fold** k.
   c. Use the classifier to predict the class labels **for the samples in fold** k.
3. The error estimates from step 2(c) are then **accumulated over all** K **folds**, to produce the cross-validation estimate of prediction error $\mathrm{CV}(\hat{r}, K, \mathbf{m})$ for each model $\mathbf{m} \in \mathcal{M}$.
4. Best data-driven model: $\hat{\mathbf{m}} \equiv \mathrm{argmin}_{\mathbf{m} \in \mathcal{M}} \mathrm{CV}(\hat{r}, K, \mathbf{m})$ leads to the prediction error, *e.g.*, $\mathrm{CV}(\hat{r}, K, \hat{\mathbf{m}})$, of the final model.

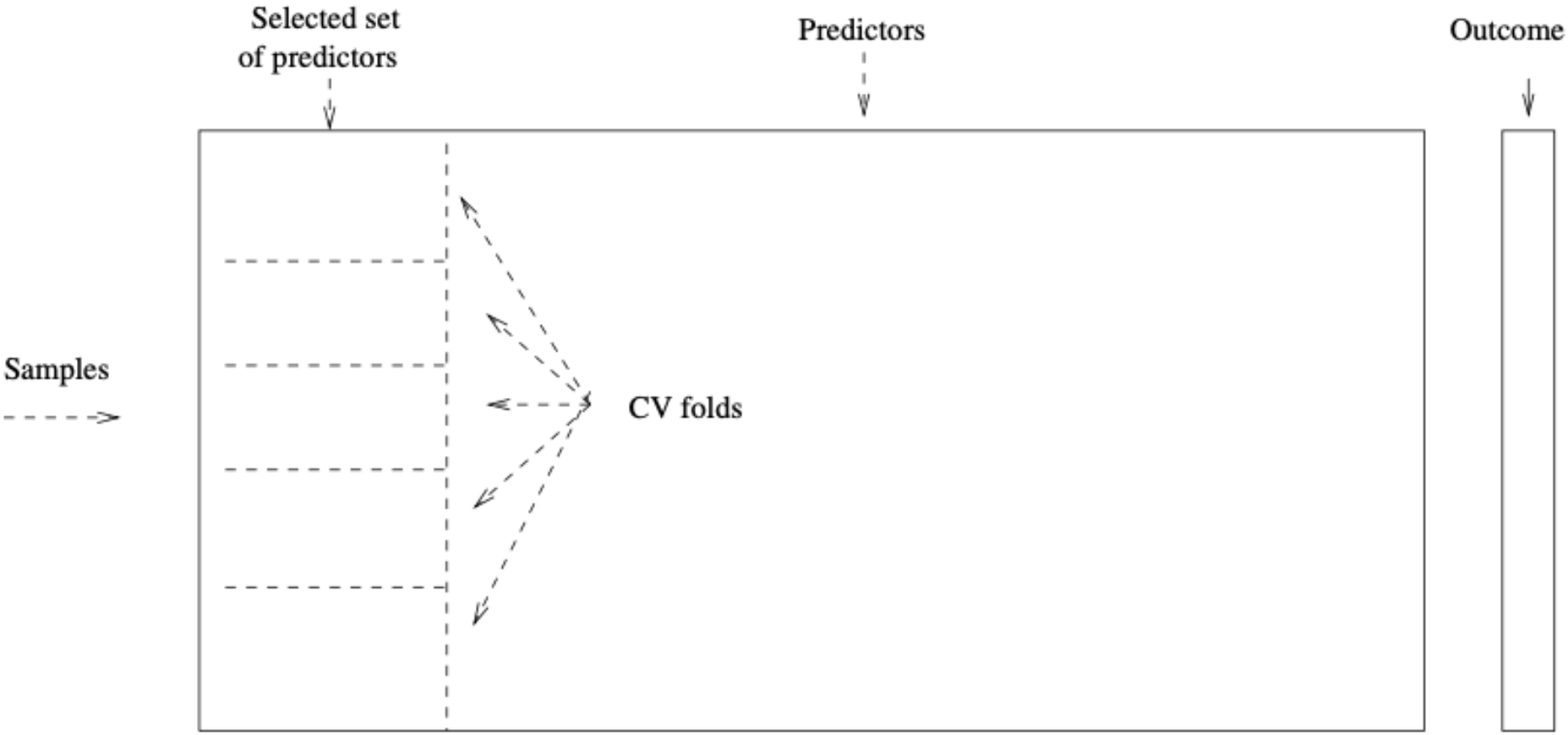


Figure 2: Cross-validation: the wrong path diagram [James et al., 2021, Figure 7.5].

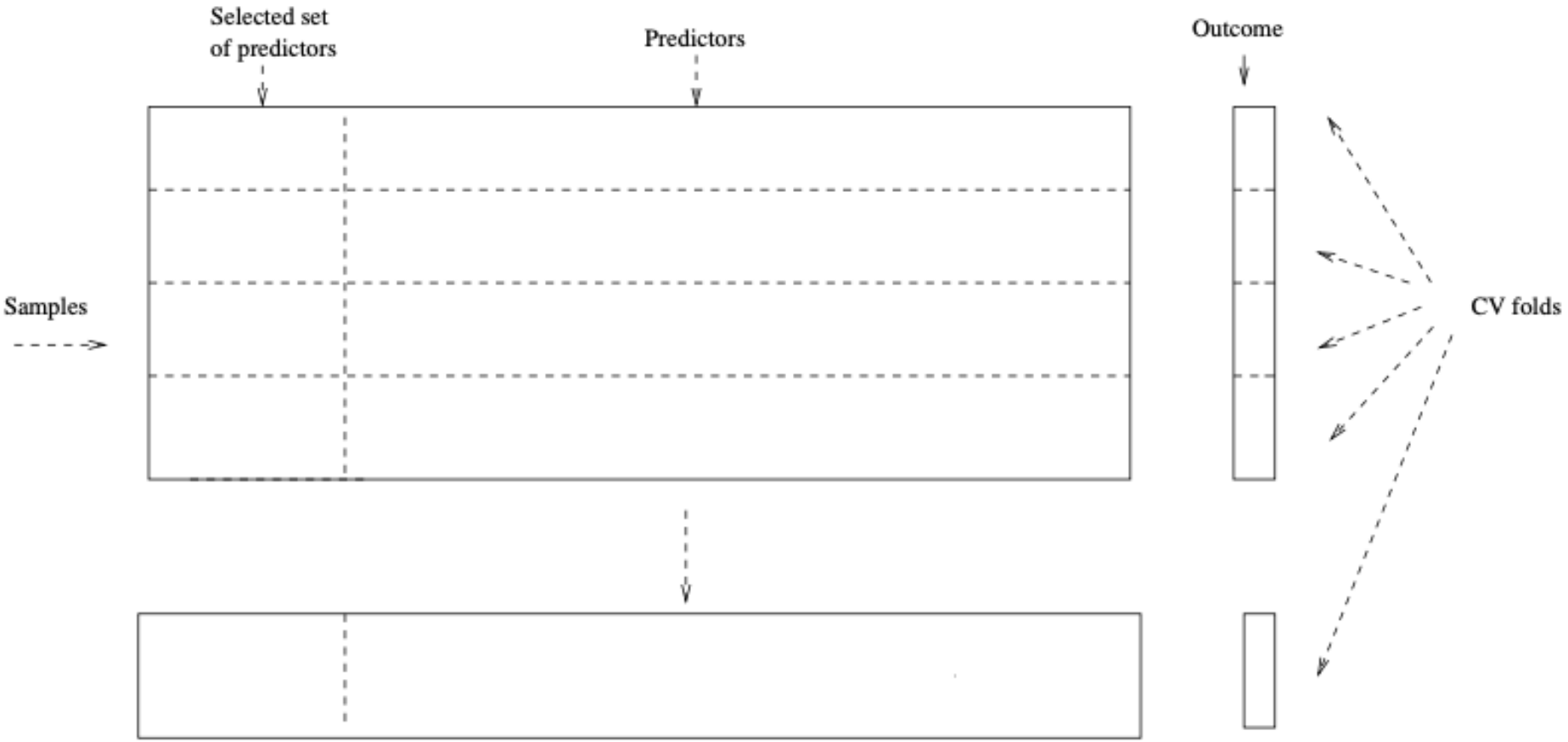


Figure 3: Cross-validation: the true path diagram [James et al., 2021, Figure 7.5].

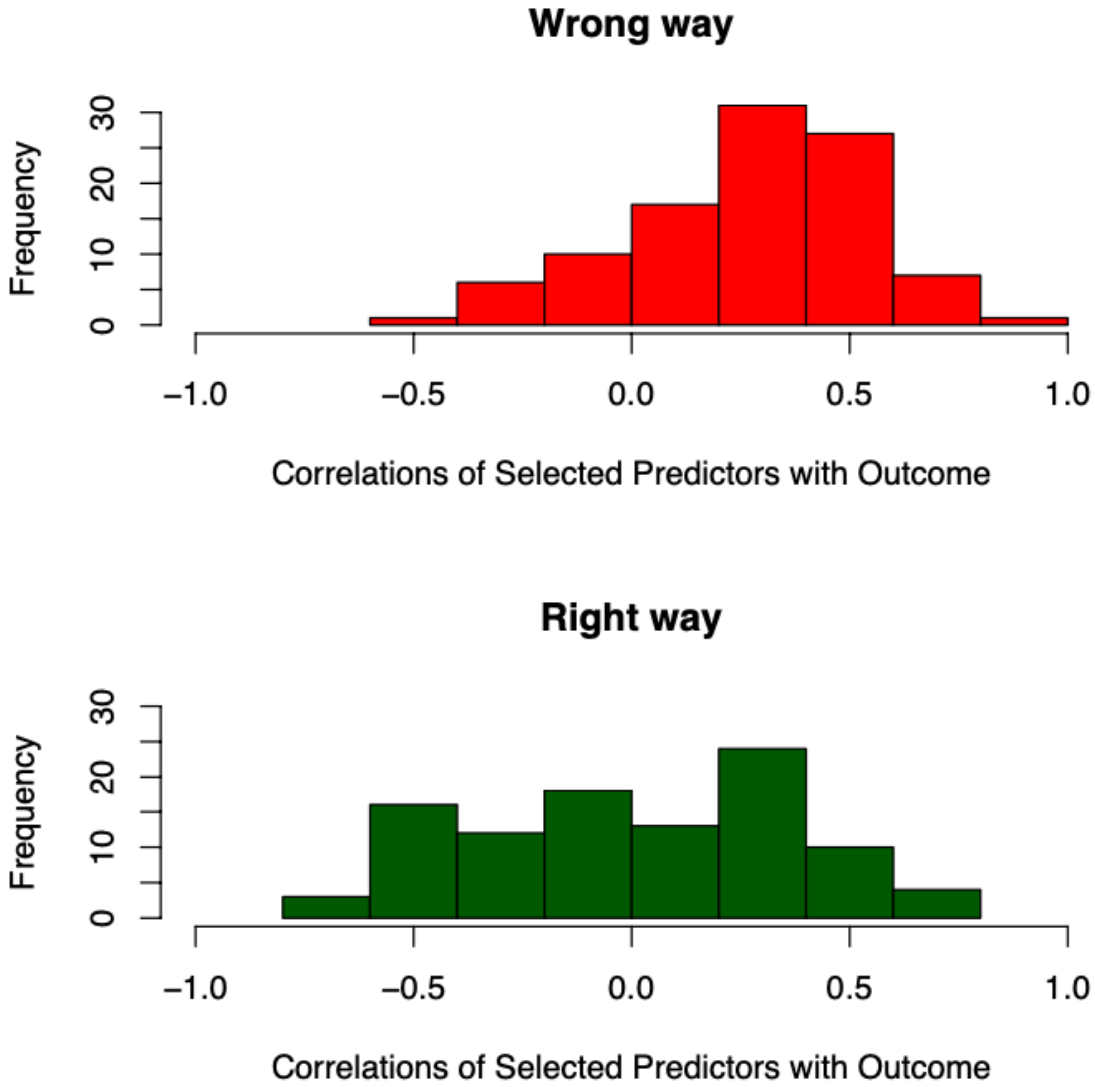


Figure 4: Cross-validation the wrong and right way: histograms shows the correlation of class labels, in 10 randomly chosen samples, with the 100 predictors chosen using the incorrect (upper red) and correct (lower **green**) versions of cross-validation [Hastie et al., 2009, Figure 7.5].

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Cross-Validation in multistep modeling procedure

- In multistep modeling procedure, cross-validation must be applied to the entire sequence of modeling steps.
- Samples must be "left out" before any selection or filtering steps are applied.
- Exceptional case: **Initial unsupervised screening steps can be done before samples are left out**.
  For example, we could select the 1000 predictors with highest variance across all 50 samples (using PCA & PCR as in CM4), before starting cross-validation.

❓ **BUT WHY**:

[9] Ambroise, C. and McLachlan, G. (2002). "Selection bias in gene extraction on the basis of microarray gene-expression data", Proceedings of the National Academy of Sciences.

Cross-Validation in multistep modeling procedure

- In multistep modeling procedure, cross-validation must be applied to the entire sequence of modeling steps.
- Samples must be "left out" before any selection or filtering steps are applied.
- Exceptional case: **Initial unsupervised screening steps can be done before samples are left out**.
  For example, we could select the 1000 predictors with highest variance across all 50 samples (using PCA & PCR as in CM4), before starting cross-validation.
- **BUT WHY**: Since this filtering does not involve the class labels, it does not give the predictors an unfair advantage. See Ambroise and McLachlan (2002)[9] for a detailed discussion of this issue.

[9] Ambroise, C. and McLachlan, G. (2002). "Selection bias in gene extraction on the basis of microarray gene-expression data", Proceedings of the National Academy of Sciences.

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Previous episode: methods for linear model selection and regularization

Subset Selection: by best subset or stepwise selection of $\mathcal{P} \equiv$ all possible subset models of P predictors, $\text{card}(\mathcal{P}) = 2^P$.

[10] Pearson, K (1894). "Contributions to the mathematical theory of evolution". Philosophical Transactions of the Royal Society of London.

Previous episode: methods for linear model selection and regularization

Subset Selection: by best subset or stepwise selection of $\mathcal{P} \equiv$ all possible subset models of P predictors, $\text{card}(\mathcal{P}) = 2^P$.

1. **Define a suitable collection of model** $\mathcal{M} = \{m : m \in \mathcal{P}\} \subset \mathcal{P}$ **and fit models** using LS, maximum likelihood estimation (MLE) or method of moments[10].

[10] Pearson, K (1894). "Contributions to the mathematical theory of evolution". Philosophical Transactions of the Royal Society of London.

Previous episode: methods for linear model selection and regularization

Subset Selection: by best subset or stepwise selection of $\mathcal{P} \equiv$ all possible subset models of P predictors, $\text{card}(\mathcal{P}) = 2^P$.

1. **Define a suitable collection of model** $\mathcal{M} = \{m : m \in \mathcal{P}\} \subset \mathcal{P}$ **and fit models** using LS, maximum likelihood estimation (MLE) or method of moments[10].
2. **Identify the best model** $\widehat{m} \in \mathcal{M}$ **via suitable model selection criteria**. (Pedro talked about this in CM3).

[10] Pearson, K (1894). "Contributions to the mathematical theory of evolution". Philosophical Transactions of the Royal Society of London.

Previous episode: methods for linear model selection and regularization

Subset Selection: by best subset or stepwise selection of $\mathcal{P} \equiv$ all possible subset models of P predictors, $\text{card}(\mathcal{P}) = 2^P$.

1. **Define a suitable collection of model $\mathcal{M} = \{m : m \in \mathcal{P}\} \subset \mathcal{P}$ and fit models** using LS, maximum likelihood estimation (MLE) or method of moments[10].
2. **Identify the best model $\widehat{m} \in \mathcal{M}$ via suitable model selection criteria**. (Pedro talked about this in CM3).

1. **Dimension Reduction:** project the P predictors into a lower dimensional subspace, *e.g.*, principal components regression, partial least squares. (In CM4, we learned how to do this).

[10] Pearson, K (1894). "Contributions to the mathematical theory of evolution". Philosophical Transactions of the Royal Society of London.

Previous episode: methods for linear model selection and regularization

3. **Shrinkage Methods:**
   - **Fit a model involving all P predictors**, using a technique that **constrains** or **regularizes** the coefficient estimates, or equivalently, that the **estimated coefficients are shrunken towards zero**.

[11] Hoerl, A. E., & Kennard, R. W. (1970). "Ridge Regression: Biased Estimation for Nonorthogonal Problems", Technometrics.

[12] Tibshirani, R. (1996). "Regression Shrinkage and Selection via the lasso". JRSS. Series B.

[13] Santosa, F., & Symes, W. W. (1986). "Linear inversion of band-limited reflection seismograms". SIAM journal on scientific and statistical computing.

Previous episode: methods for linear model selection and regularization

3. **Shrinkage Methods:**
   - **Fit a model involving all P predictors**, using a technique that **constrains** or **regularizes** the coefficient estimates, or equivalently, that the **estimated coefficients are shrunken towards zero**.

   Not immediately obvious why such a constraint should improve the fit.

   This shrinkage (also known as **regularization**) has the effect of reducing variance and can also perform **variable selection**.

[11] Hoerl, A. E., & Kennard, R. W. (1970). "Ridge Regression: Biased Estimation for Nonorthogonal Problems", Technometrics.

[12] Tibshirani, R. (1996). "Regression Shrinkage and Selection via the lasso". JRSS. Series B.

[13] Santosa, F., & Symes, W. W. (1986). "Linear inversion of band-limited reflection seismograms". SIAM journal on scientific and statistical computing.

Previous episode: methods for linear model selection and regularization

3. **Shrinkage Methods:**
   - **Fit a model involving all P predictors**, using a technique that **constrains** or **regularizes** the coefficient estimates, or equivalently, that the **estimated coefficients are shrunken towards zero**.

Not immediately obvious why such a constraint should improve the fit.

This shrinkage (also known as **regularization**) has the effect of reducing variance and can also perform **variable selection**.

The two best-known techniques: ridge regression[11] and Lasso [12] [13].

[11] Hoerl, A. E., & Kennard, R. W. (1970). "Ridge Regression: Biased Estimation for Nonorthogonal Problems", Technometrics.

[12] Tibshirani, R. (1996). "Regression Shrinkage and Selection via the lasso". JRSS. Series B.

[13] Santosa, F., & Symes, W. W. (1986). "Linear inversion of band-limited reflection seismograms". SIAM journal on scientific and statistical computing.

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

Dimension reduction methods

- The methods that we have discussed so far in this chapter have involved fitting linear regression models, via least squares or a shrunken approach, using the **original predictors**, $X_1, X_2, \ldots, X_P$.

Dimension reduction methods

- The methods that we have discussed so far in this chapter have involved fitting linear regression models, via least squares or a shrunken approach, using the **original predictors**, $X_1, X_2, \ldots, X_P$.
- We now explore a class of approaches that transform the predictors and then fit a least squares model using the transformed variables. We will refer to these techniques as dimension reduction methods.

Dimension reduction methods: details

- Let $Z_1, \ldots, Z_Q$ represent $Q < P$ linear combinations of our original predictors. That is, for each $q \in [Q]$, it holds that

$$Z_q = \sum_{p=1}^{P} W_{qp} X_p, \text{ for some matrices } \mathbf{W} \equiv (W_{qp})_{q \in [Q], p \in [P]}. \tag{6}$$

Dimension reduction methods: details

- Let $Z_1, \ldots, Z_Q$ represent $Q < P$ linear combinations of our original predictors. That is, for each $q \in [Q]$, it holds that

$$Z_q = \sum_{p=1}^{P} W_{qp} X_p, \text{ for some matrices } \mathbf{W} \equiv (W_{qp})_{q \in [Q], p \in [P]}. \tag{6}$$

- Via using OLS, we can then fit the following linear regression model

$$y_n = \gamma_0 + \sum_{q=1}^{Q} \gamma_q z_{nq} + \epsilon_n, n \in [N], \quad \boldsymbol{\gamma} = (\gamma_0, \gamma_1, \ldots, \gamma_Q)$$

$$= \gamma_0 + \sum_{q=1}^{Q} \gamma_q \sum_{p=1}^{P} W_{qp} X_p + \epsilon_n = \gamma_0 + \sum_{p=1}^{P} \underbrace{\sum_{q=1}^{Q} \gamma_q W_{qp}}_{\beta_p} X_p + \epsilon_n. \tag{7}$$

Dimension reduction methods: details

- Let $Z_1, \ldots, Z_Q$ represent $Q < P$ linear combinations of our original predictors. That is, for each $q \in [Q]$, it holds that

$$Z_q = \sum_{p=1}^{P} W_{qp} X_p, \text{ for some matrices } \mathbf{W} \equiv (W_{qp})_{q \in [Q], p \in [P]}. \tag{6}$$

- Via using OLS, we can then fit the following linear regression model

$$y_n = \gamma_0 + \sum_{q=1}^{Q} \gamma_q z_{nq} + \epsilon_n, n \in [N], \quad \boldsymbol{\gamma} = (\gamma_0, \gamma_1, \ldots, \gamma_Q)$$

$$= \gamma_0 + \sum_{q=1}^{Q} \gamma_q \sum_{p=1}^{P} W_{qp} X_p + \epsilon_n = \gamma_0 + \sum_{p=1}^{P} \underbrace{\sum_{q=1}^{Q} \gamma_q W_{qp}}_{\beta_p} X_p + \epsilon_n. \tag{7}$$

Note that in model (7), of as a special case of the original linear regression model via constraining the estimated β_p coefficients win in the bias-variance tradeoff!

Dimension reduction methods: details

- Let $Z_1, \ldots, Z_Q$ represent $Q < P$ linear combinations of our original predictors. That is, for each $q \in [Q]$, it holds that

$$Z_q = \sum_{p=1}^{P} W_{qp} X_p, \text{ for some matrices } \mathbf{W} \equiv (W_{qp})_{q \in [Q], p \in [P]}. \tag{6}$$

- Via using OLS, we can then fit the following linear regression model

$$y_n = \gamma_0 + \sum_{q=1}^{Q} \gamma_q z_{nq} + \epsilon_n, n \in [N], \quad \boldsymbol{\gamma} = (\gamma_0, \gamma_1, \ldots, \gamma_Q)$$

$$= \gamma_0 + \sum_{q=1}^{Q} \gamma_q \sum_{p=1}^{P} W_{qp} X_p + \epsilon_n = \gamma_0 + \sum_{p=1}^{P} \underbrace{\sum_{q=1}^{Q} \gamma_q W_{qp}}_{\beta_p} X_p + \epsilon_n. \tag{7}$$

Note that in model (7), of as a special case of the original linear regression model via constraining the estimated β_p coefficients win in the bias-variance tradeoff!
If $\mathbf{W}$ are chosen wisely, *e.g.*, using PCA from CM4 to have Principal Components Regression, then such dimension reduction approaches can often outperform OLS regression!

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

An application to the credit data

- **Description:** the response is balance (average credit card debt for 400 individuals) and there are 6 quantitative predictors: income (in thousands of dollars), limit (credit limit), rating (credit rating), cards (number of credit cards), age, education (years of education), and 4 qualitative variables: own (house ownership), student (student status), married (Yes or No), and region (East, West or South). [James et al., 2021, Section 3.3].

An application to the credit data

- **Description:** the response is balance (average credit card debt for 400 individuals) and there are 6 quantitative predictors: income (in thousands of dollars), limit (credit limit), rating (credit rating), cards (number of credit cards), age, education (years of education), and 4 qualitative variables: own (house ownership), student (student status), married (Yes or No), and region (East, West or South). [James et al., 2021, Section 3.3].
- **Goal:** develop an accurate model that can be used to **predict balance** on the basis of 10 predictors $\leftarrow$ Using glmnet package in $\boldsymbol{R}$.

```
> head(Credit)                                                          Y
  Income Limit Rating Cards Age Education Own Student Married Region Balance
1 14.891  3606    283     2  34        11  No      No     Yes  South     333
2 106.025 6645    483     3  82        15 Yes     Yes     Yes   West     903
3 104.593 7075    514     4  71        11  No      No      No   West     580
4 148.924 9504    681     3  36        11 Yes     No      No   West     964
5 55.882  4897    357     2  68        16  No      No     Yes  South     331
6 80.180  8047    569     4  77        10  No      No      No  South    1151
```

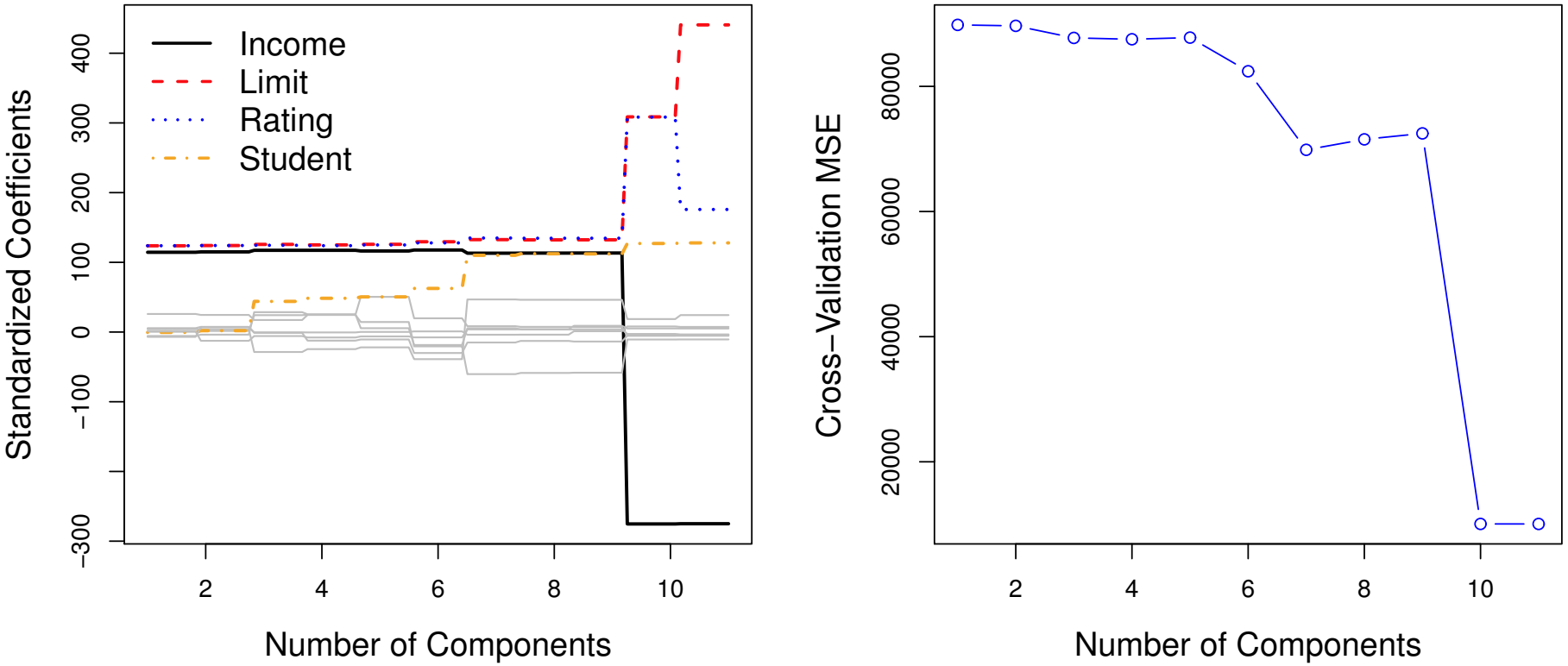


Figure 5: **Choosing the number of directions Q via cross-validation** on the Credit data. Left: PCR standardized coefficient estimates on the Credit data set for different values of Q. Right: The ten-fold cross-validation MSE obtained using PCR, as a function of Q [James et al., 2021, Figure 6.20].

Outline

1. Previous Episode: Cross-Validation in linear regression
   - Training error versus test error
   - Test error in linear regression
   - The general model selection paradigm
   - K-Fold cross-validation
2. Cross-Validation: Right and Wrong Way
   - Test error in binary (supervised) classification
   - Cross-Validation for classification problems
   - Cross-Validation: right and wrong
   - Cross-Validation in multistep modeling procedure
3. Cross-Validation for Principal Components Regression
   - Linear model selection and regularization
   - Dimension reduction methods
   - An application to the credit data
   - An application to the prostate cancer

An application to the prostate cancer

- **Description:** represent the correlation between the **level of prostate specific antigen** (PSA) and a number of **clinical measures**, in 97 men who were about to receive a radical prostatectomy[14].

[14] Stamey et al. (1989). "Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate II radical prostatectomy treated patients", Journal of Urology.

An application to the prostate cancer

- **Description:** represent the correlation between the **level of prostate specific antigen** (PSA) and a number of **clinical measures**, in 97 men who were about to receive a radical prostatectomy[14].
- **Goal:** predict the log of PSA (lpsa) from a number of measurements including log cancer volume (lcavol), log prostate weight (lweight), age, log of benign prostatic hyperplasia amount (lbph), seminal vesicle invasion (svi), log of capsular penetration (lcp), Gleason score (gleason), and percent of Gleason scores 4 or 5 (pgg45).

```
> head(df)
     lcavol  lweight age      lbph svi       lcp gleason pgg45       lpsa
1 -0.5798185 2.769459  50 -1.386294   0 -1.386294       6     0 -0.4307829
2 -0.9942523 3.319626  58 -1.386294   0 -1.386294       6     0 -0.1625189
3 -0.5108256 2.691243  74 -1.386294   0 -1.386294       7    20 -0.1625189
4 -1.2039728 3.282789  58 -1.386294   0 -1.386294       6     0 -0.1625189
5  0.7514161 3.432373  62 -1.386294   0 -1.386294       6     0  0.3715636
6 -1.0498221 3.228826  50 -1.386294   0 -1.386294       6     0  0.7654678
```

[14] Stamey et al. (1989). "Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate II radical prostatectomy treated patients", Journal of Urology.

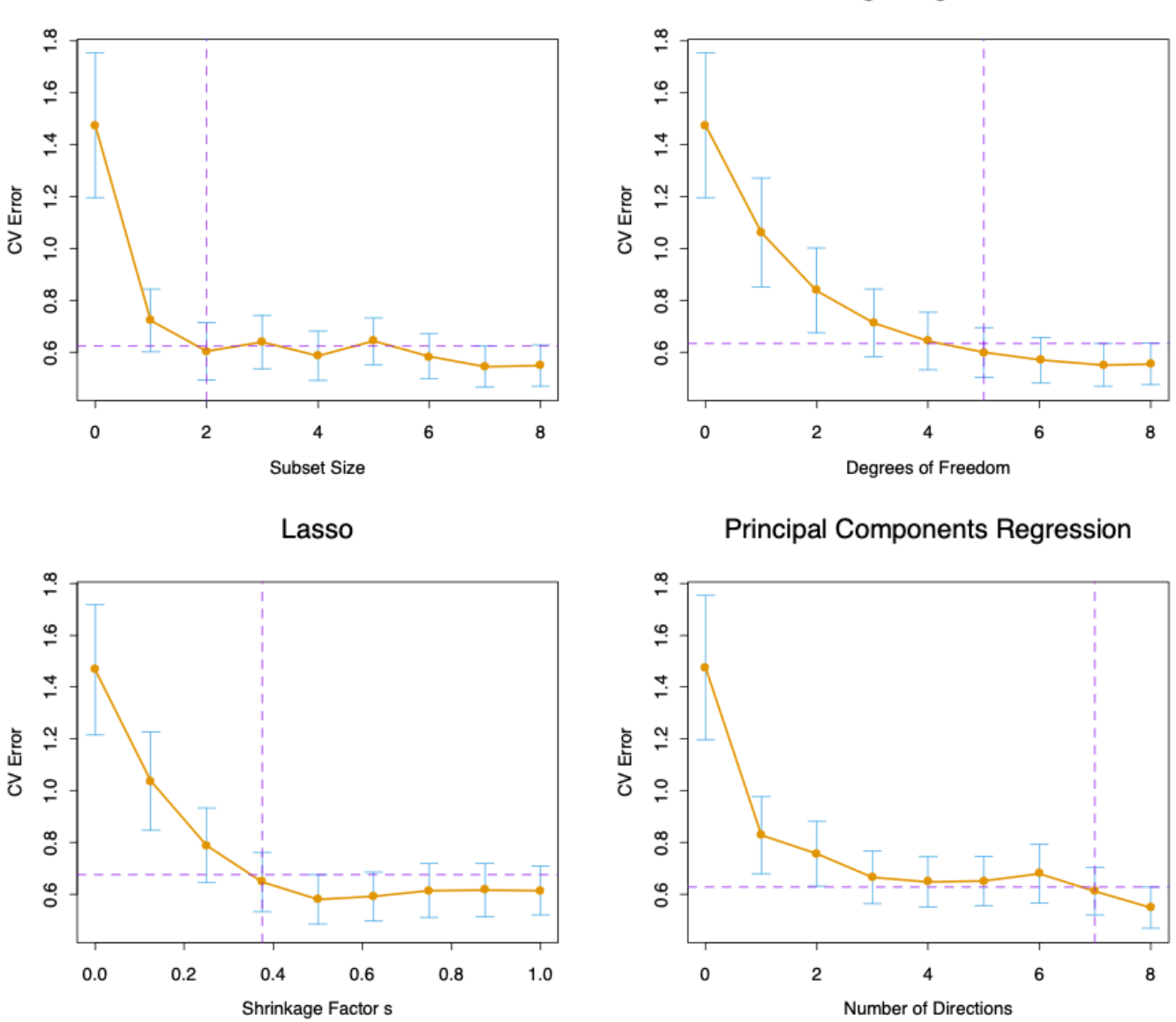


Figure 6: **Estimated prediction error curves and their standard errors for the various selection and shrinkage methods** on the Prostate Cancer data. The estimates of prediction error and their standard errors were obtained by 10-folds cross-validation. [Hastie et al., 2009, Figure 3.7].

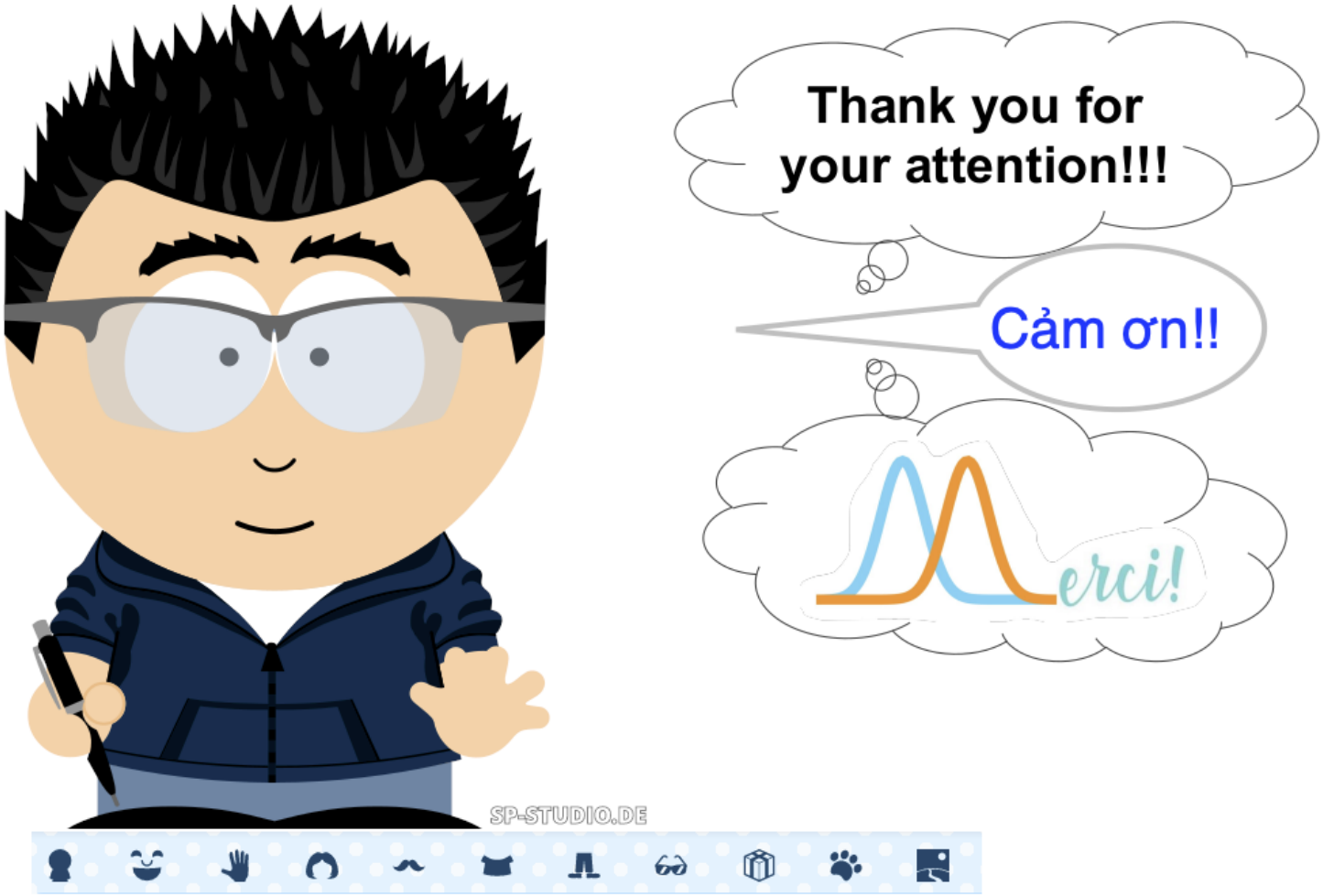
Term	LS	Best Subset	Ridge	Lasso	PCR	PLS
Intercept	2.465	2.477	2.452	2.468	2.497	2.452
lcavol	0.680	0.740	0.420	0.533	0.543	0.419
lweight	0.263	0.316	0.238	0.169	0.289	0.344
age	−0.141		−0.046		−0.152	−0.026
lbph	0.210		0.162	0.002	0.214	0.220
svi	0.305		0.227	0.094	0.315	0.243
lcp	−0.288		0.000		−0.051	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

Figure 7: Estimated coefficients and test error results, for different subset and shrinkage methods applied to the prostate data. The blank entries correspond to variables omitted [Hastie et al., 2009, Table 3.3].

Each method has a **complexity parameter**, and this was chosen to minimize an estimate of prediction error based on **model selection criteria, *e.g.,*** 10-**folds cross-validation.**

Original data (97) = training set (67, cross-validation) + test set (30, juge performance of the selected model).

"Essentially, all models are wrong, but some are useful".[15]



⬆ This is my best data-driven model to approximate myself.

[15] Box, G. E.P. (1979). "Robustness in the strategy of scientific model building". In Robustness in Statistics (pp. 201-236). Academic Press.

References

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2 of *Springer Texts in Statistics*. Springer.

(Cited on pages 58, 59, 60, 61, 62, 71, 98, and 99.)

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning: with Applications in R*, volume 2 of *Springer Texts in Statistics*. Springer.

(Cited on pages 27, 69, 70, 92, 93, and 94.)