Computer Intensive Statistics and Applications
Chapter 4: Simulation-Based Methods

Shaobo Jin

Department of Mathematics

General Setup

The expectation and maximization (EM) algorithm is one of the most used methods in statistics. The general setup is

- we can specify a statistical model for a random vector X, say, its density is $f(x \mid \theta)$,
- we observe some data y, corresponding to the random variable Y,
- the observed y is a function of x, as $y = y(x)$, a many-to-one mapping.

In this setting, X is our complete data, Y is our incomplete data, and the density of y satisfies

$$g(y \mid \theta) = \int_{\{x;\, y(x)=y\}} f(x \mid \theta)\, dx.$$

Typical Applications: Missing Value

A typical example of such setup is the missing value problem.

- In the ideal world, we could have observed x, e.g., all responses to a questionnaire,
- In reality, we only observe y, e.g., some questions are not answered or some people choose not to participate.

Typical Applications: Clustering

Clustering is also a common problem.

- We know our data come from several populations as a mixture of distributions, but we don't know which group each observation comes from.
- Let Z be a multinomial random variable that indicates which group the observations comes from:

$$P(Z = k) = p_k \geq 0.$$

- Within group k $(Z = k)$, our data follow some distribution with density function $f_k(y)$.
- Then, the observation vector for a single object is modeled as a finite mixing distribution $\sum_{k=1}^{K} p_k f_k(y)$.

Complete Likelihood and Observed Likelihood

If x were observed, then the likelihood function is

$$L_C(\theta) = f(x \mid \theta).$$

However, we only observe y in practice, the likelihood is

$$L(\theta) = g(y \mid \theta) = \int_{\{x;\, y(x)=y\}} f(x \mid \theta)\, dx.$$

L_C is known as the complete likelihood and L is known as the observed likelihood. The maximum likelihood estimators of unknown parameters maximize the observed likelihood.

EM Algorithm

However, directly maximizing the observed likelihood is computationally not easy. The EM algorithm is often used instead.

Algorithm 1: EM Algorithm

Suppose that the parameter vector is θ. Obtain initial guess $\theta^{(0)}$;

while *at step* t **do**

  E step: Find conditional expectation $\mathrm{E}\left[\log f\left(x \mid \theta\right) \mid y, \hat{\theta}^{(t-1)}\right]$ where the expectation is done to the conditional distribution of $x \mid y$ and the parameter value is $\hat{\theta}^{(t-1)}$;

  M step: Maximize the conditional expectation with respect to θ and obtain the updated value $\boldsymbol{\theta}^{(t)}$;

end

Kullback-Leilber Divergence

Theorem (Theorem 4.1, Kullback-Leibler)

Let p and g be two densities, then

$$E_p\left[\log p\left(X\right)\right] \quad \geq \quad E_p\left[\log g\left(X\right)\right],$$

where the subscript indicates the true distribution of X.

We often call

$$\mathrm{KL}\left(p, g\right) \quad = \quad \mathrm{E}_p\left[\log\left(\frac{p\left(X\right)}{g\left(X\right)}\right)\right] \geq 0$$

the Kullback-Leibler divergence. A large $\mathrm{KL}\left(p, g\right)$ means that g is far away from p.

Conditional Expectation

In the literature, the conditional expectation

$$Q\left(\theta \mid \theta'\right) \quad = \quad \mathrm{E}\left[\log f\left(x \mid \theta\right) \mid y, \theta'\right]$$

is often called the surrogate function.

Theorem (Theorem 4.2)

The surrogate function satisfies

$$Q\left(\theta' \mid \theta'\right) - \log g\left(y \mid \theta'\right) \quad \geq \quad Q\left(\theta \mid \theta'\right) - \log g\left(y \mid \theta\right).$$

Why Does EM Algorithm Work?

Theorem 4.2 implies that

$$Q\left(\theta^{(t-1)} \mid \theta^{(t-1)}\right) - \log g\left(y \mid \theta^{(t-1)}\right) \geq Q\left(\theta \mid \theta^{(t-1)}\right) - \log g\left(y \mid \theta\right).$$

The M-step in the EM algorithm maximizes $Q\left(\theta \mid \theta^{(t-1)}\right)$ as a function of θ. Hence,

$$\begin{aligned} \log g\left(y \mid \theta^{(t)}\right) &\geq Q\left(\theta^{(t)} \mid \theta^{(t-1)}\right) - Q\left(\theta^{(t-1)} \mid \theta^{(t-1)}\right) + \log g\left(y \mid \theta^{(t-1)}\right) \\ &\geq \log g\left(y \mid \theta^{(t-1)}\right). \end{aligned}$$

That is, the observed likelihood increases as we maximize the conditional expectation.

EM Algorithm: Example

The most common model is the Gaussian mixture.

EM Algorithm For Gaussian Mixture

We know that

$$\begin{aligned} Y \mid Z = k &\sim N\left(\mu_k, \sigma_k^2\right), \\ p_k &= P\left(Z = k\right), \quad k = 1, ..., K. \end{aligned}$$

We observed a random sample y_1, ..., y_n, but we do not observe Z. Estimate p_k, μ_k and σ_k^2.

EM Algorithm: Missing Example

EM Algorithm For Missing Value

Suppose that we have n observations from a bivariate normal distribution $N\left(0, \begin{bmatrix} \sigma & \rho\sigma \\ \rho\sigma & \sigma \end{bmatrix}\right)$, where ρ and σ are the unknown parameters. We observe

$$Y = \begin{bmatrix} Y_{11} & Y_{12} \\ - & Y_{22} \end{bmatrix},$$

where Y_{21} is missing. For a bivariate normal distribution

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim N\left(0, \begin{bmatrix} \sigma & \rho\sigma \\ \rho\sigma & \sigma \end{bmatrix}\right),$$

we know that $X_1 \mid X_2 = x_2 \sim N\left[\rho x_2,\ \left(1-\rho^2\right)\sigma\right]$.

Other Details

- We need to determine K. One way is to use the information criterion

$$\begin{aligned} \text{AIC} &= -2 \log g\left(y \mid \hat{\theta}\right) + 2 \times \text{number of parameters}, \\ \text{BIC} &= -2 \log g\left(y \mid \hat{\theta}\right) + \log n \times \text{number of parameters}. \end{aligned}$$

- It is often the case that some errors (e.g., non-convergence) are encountered for complicated models.
- EM algorithm is prone to local maxima. Hence, it is common to use multiple starting values.
- The EM algorithm can become extremely slow after a few steps. Various acceleration techniques have been proposed.

Self-Consistency of EM Algorithm

If $\theta^{(t)}$ maximizes $Q\left(\theta \mid \theta^{(t-1)}\right)$ as a function of θ, then the observed likelihood increases, i.e.,

$$\log g\left(y \mid \theta^{(t)}\right) \quad \geq \quad \log g\left(y \mid \theta^{(t-1)}\right).$$

- Suppose that $\hat{\theta}$ maximizes $\log g\left(y \mid \theta\right)$, i.e., $\hat{\theta}$ is the MLE.
- The MLE must satisfy $Q\left(\hat{\theta} \mid \hat{\theta}\right) \geq Q\left(\theta \mid \hat{\theta}\right)$ for all θ.
  - Otherwise $\log g\left(y \mid \theta\right) \geq \log g\left(y \mid \hat{\theta}\right)$ for some θ.
- Hence, MLE is one of the local maxima.

MCEM Algorithm

It is not always easy to obtain a closed form expression when we evaluate the expectation

$$Q\left(\theta \mid \hat{\theta}^{(t-1)}\right) = \mathrm{E}\left[\log f\left(x \mid \theta\right) \mid y, \hat{\theta}^{(t-1)}\right].$$

In the case where $x = (y, z)$, where z is unobserved, we can often approximate $Q\left(\theta \mid \theta'\right)$ using Monte Carlo integration. The resulting EM algorithm is known as the MCEM algorithm.

In the E step,

- Generate independently z_1, ..., z_M from $p\left(z \mid y, \hat{\theta}^{(t-1)}\right)$.
- Approximate $Q\left(\theta \mid \hat{\theta}^{(t-1)}\right)$ by

$$Q\left(\theta \mid \hat{\theta}^{(t-1)}\right) = \frac{1}{M}\sum_{m=1}^{M} \log f\left(y, z_j \mid \theta\right).$$

Stochastic Nature

Since a Monte Carlo approximation error is introduced at the E step, the monotonicity property of the EM algorithm is lost.

- $\hat{\theta}^{(t)}$ will fluctuate around some value as the algorithm moves closer to convergence.
- Specifying a suitable M and monitoring convergence become very important.
- It is often recommended to use small values of M in the initial stages and increase M as the algorithm moves forward.
- To monitor convergence, one easy tool is to plot $\hat{\theta}^{(t)}$ against the steps. Convergence is indicated by stable and random fluctuation about some value.
- A more sophisticated approach is to derive a confidence interval for $Q\left(\hat{\theta}^{(t-1)} \mid \hat{\theta}^{(t-1)}\right) - Q\left(\hat{\theta}^{(t)} \mid \hat{\theta}^{(t-1)}\right)$ and terminate the algorithm if the upper limit of the confidence interval is below some tolerance level.

Score Function

Let

$$S_C(\theta) = \frac{\partial \log f(x \mid \theta)}{\partial \theta}, \qquad S(\theta) = \frac{\partial \log g(y \mid \theta)}{\partial \theta}$$

be the complete score function and the observed score function, respectively. Then, if we can change the order of integration and differentiation, we have

$$S(\theta) = \mathrm{E}\left[S_C(\theta) \mid Y\right].$$

We can also define the Fisher information:

$$\text{observed}: \quad \mathcal{I}_{\text{obs}} = -\frac{\partial^2 \log g(y \mid \theta)}{\partial \theta \partial \theta^T},$$

$$\text{expected}: \quad \mathcal{I}_{\text{exp}} = \mathrm{var}\left[S(\theta)\right] = -\mathrm{E}\left[\frac{\partial^2 \log g(y \mid \theta)}{\partial \theta \partial \theta^T}\right].$$

Quantify Uncertainty

Since the solution of the EM algorithm converges to the MLE, it is also a random variable and we need to quantity its uncertainty.

- For MLE, the Fisher information is often used to compute the standard errors of the MLE.
- Under some conditions the MLE $\hat{\theta}$ satisfies

$$\mathcal{I}^{1/2}\left(\hat{\theta}-\theta\right) \quad \xrightarrow{d} \quad N\left(0, I\right).$$

- Hence, the standard errors can be obtained as the square root of the diagonal entries of $\mathcal{I}_{\text{obs}}\left(\hat{\theta}\right)$ or $\mathcal{I}_{\text{exp}}\left(\hat{\theta}\right)$.
- Confidence intervals can be constructed accordingly.

Approximate Information Matrix

1. Sometimes we can compute $S_i(\theta) = \frac{\partial \log g(y_i|\theta)}{\partial \theta}$. Then, we can approximate the Fisher information by

$$\hat{\mathcal{I}} = \frac{1}{n}\sum_{i=1}^{n} \frac{\partial \log g(y_i \mid \theta)}{\partial \theta} \frac{\partial \log g(y_i \mid \theta)}{\partial \theta^T}.$$

2. Sometimes it is not straightforward to compute $S_i(\theta)$. The Louis method means that

$$\mathcal{I}_{\text{obs}} = -\mathrm{E}\left[\frac{\partial^2 \log f(x \mid \theta)}{\partial \theta \partial \theta^T} \mid Y\right] - \mathrm{E}\left[S_C(\theta) S_C^T(\theta) \mid Y\right] + S(\theta) S^T(\theta).$$

If we take $\hat{\theta}$ as the MLE, then $S\left(\hat{\theta}\right) = 0$ and we skip the evaluation of observed gradient.

Measurement Error

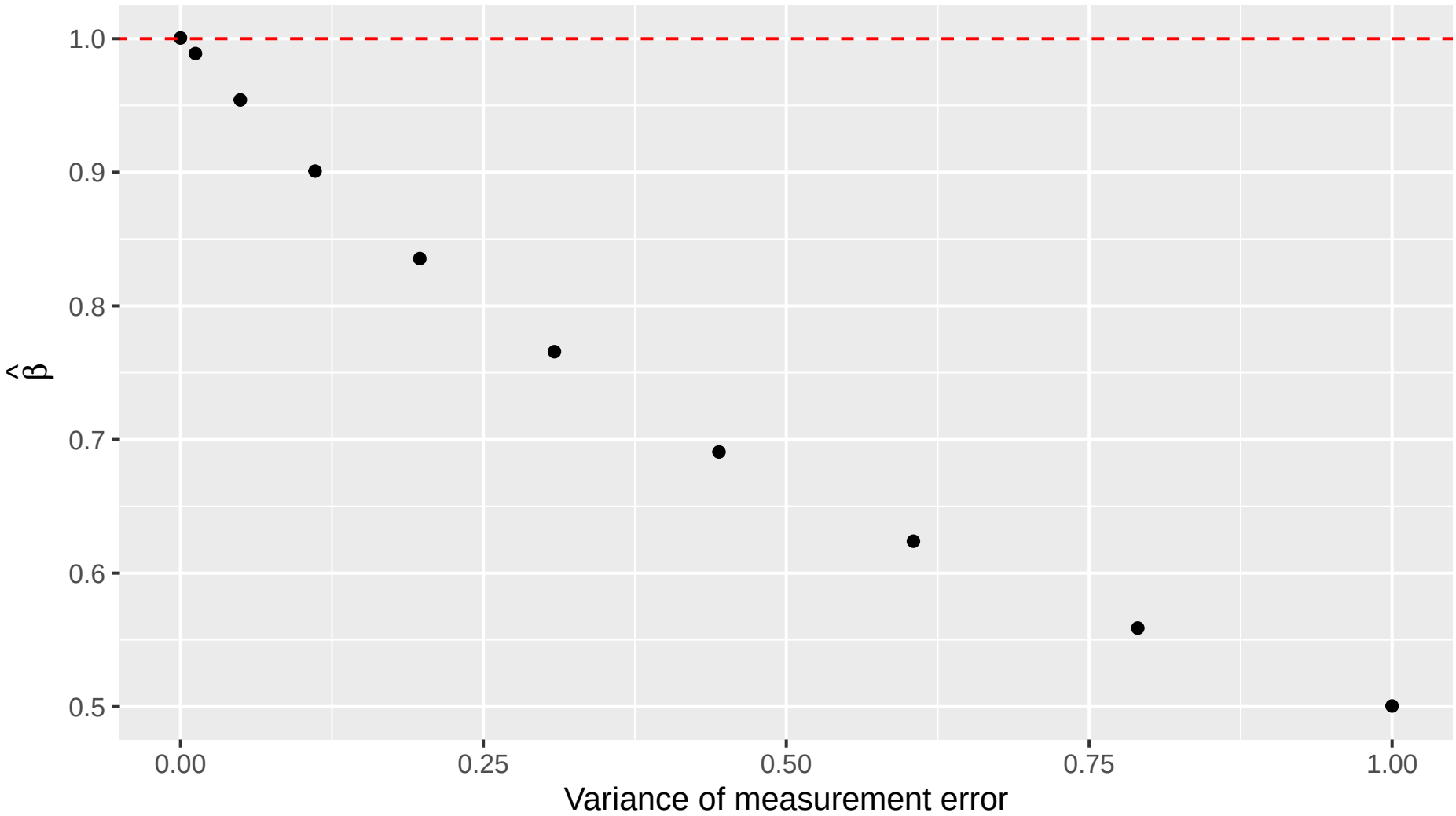
It is often the case that our data are incomplete.

- We want to model $\mathrm{E}[Y \mid U, V]$. But we cannot observe U directly.
- U is called a latent variable.

For example, in the measurement error model,

- instead of observing U, we observe $X = U + \sigma Z$, where $\mathrm{Var}(Z) = 1$ and σZ is the measurement error.
- our observed values are (Y, X, V), and U is the latent variable.
- if $T(Y, U, V)$ is a valid estimator, then $T(Y, X, V)$ is often not.

Example: Measurement Error in Regression



Simulation-Extrapolation Estimation

The simulation-extrapolation (SIMEX) estimation for known σ^2 works as follows.

Algorithm 2: SIMEX Algorithm

Choose a sequence $\{\lambda_1, ..., \lambda_K\}$, where $\lambda_k > 0$ for all k ;
for *k from 1 to K* **do**
  for *b from 1 to B* **do**
    Generate new errors $Z^*_{b,i}$ (zero mean, variance 1) ;
    Obtain new samples $X_{b,i}(\lambda) = X_i + \sqrt{\lambda_k}\sigma Z^*_i$, $\forall i$;
    Calculate $\hat{\theta}_b(\lambda) = T(\boldsymbol{Y}, \boldsymbol{V}, \boldsymbol{X}_b(\lambda_k))$;
  end
  Calculate the average $\bar{\theta}(\lambda_k) = b^{-1}\sum_{b=1}^{B}\hat{\theta}_b(\lambda_k)$;
end
Fit a curve $\hat{f}(\lambda)$ such that $\sum_{k=1}^{K}\left[\hat{f}(\lambda_k) - \bar{\theta}(\lambda_k)\right]^2$ is minimal ;
The SIMEX estimator is $\hat{\theta}_{\text{SIMEX}} = \hat{f}(-1)$

SIMEX for Linear Regression

Example

Consider the linear relationship

$$y_i = \beta_0 + \beta_1 v_i + \beta_2 z_i + \epsilon_i.$$

Instead of observing z, we observe

$$x_i = z_i + \delta_i,$$

where $\text{Var}(\delta_i) = 0.5$.

Introduction

Suppose that we have the response variable Y, and p variables (features) X_1, ..., X_p.

- We want to select the minimal set $\{X_{j_1}, ..., X_{j_m}\}$ from $\{X_1, ..., X_p\}$ to model

$$\mathrm{E}\left[Y \mid X_1, ..., X_p\right] \quad = \quad \mathrm{E}\left[Y \mid X_{j_1}, ..., X_{j_m}\right].$$

- The set of indices $\mathcal{A} = \{j_1, ..., j_m\}$ is called an active set.

We will introduce

1. backward, forward, and stepwise procedures,
2. information criterion,
3. cross validation.

Backward and Forward Selection in Linear Regression

Suppose that

$$\boldsymbol{Y} = \boldsymbol{X}\beta + \epsilon, \quad \epsilon \sim N_n\left(0\sigma^2 \boldsymbol{I}_n\right),$$

where the rank of the $n \times p$ matrix$\boldsymbol{X}$ is p. Define

$$\text{RSS}\left(\mathcal{A}\right) = \sum_{i=1}^{n}\left(y_i - \sum_{j\in\mathcal{A}} x_j\hat{\beta}_j\right)^2,$$

where our hypothesized model is $\sum_{j\in\mathcal{A}} x_j\beta_j$, and $\beta_j = 0$ for $j \notin \mathcal{A}$.

Backward and Forward Selection in Linear Regression

Suppose that $\mathcal{A}_0 \subset \mathcal{A}_1$. Then,

$$F(\mathcal{A}_0, \mathcal{A}_1) = \frac{(\text{RSS}(\mathcal{A}_0) - \text{RSS}(\mathcal{A}_1)) / (p - p_0)}{\text{RSS}(\mathcal{A}_1) / (n - p)} \quad \sim \quad F(p - p_0, n - p),$$

where $\mathcal{A}_1$ has p_0 more features than $\mathcal{A}_0$.

Using F values,

- Forward selection starts with an intercept model and sequentially adds terms.
- Backward elimination starts with a model with all features and sequentially removes terms.
- Stepwise regression is a combination of both by checking whether adding one term or deleting one term.

Backward Elimination in Linear Regression

Algorithm 3: Backward Elimination Using F Test

Specify a significance level α ;
Specify the active set $\mathcal{A}$ that contains all features ;
while *Proceed* **do**
    For each $j \in \mathcal{A}$, calculuate $F\left(\mathcal{A} \setminus \{j\}, \mathcal{A}\right)$;
    Select l such that $F\left(\mathcal{A} \setminus \{l\}, \mathcal{A}\right) = \min_{j \in \mathcal{A}} F\left(\mathcal{A} \setminus \{j\}, \mathcal{A}\right)$;
    Calculate c, where c is the $1-\alpha$ quantile of $F\left(p-1, n-p\right)$ with p being the size of the active set $\mathcal{A}$;
    if $F\left(\mathcal{A} \setminus \{l\}, \mathcal{A}\right) < c$ **then**
        $\mathcal{A} \leftarrow \mathcal{A} \setminus \{l\}$;
    else
        Select the model with active set $\mathcal{A}$ and terminate the procedure ;
    end
end

Forward Selection in Linear Regression

Algorithm 4: Forward Selection Using F Test

Specify a significance level α ;
Specify the active set $\mathcal{A} = \emptyset$ that contains no feature ;
while *Proceed* **do**
    For each $j \in \{1, ..., p\} \setminus \mathcal{A}$, Select l such that $F(\mathcal{A}, \mathcal{A} \cup \{l\}) = \max_{j \in \{1,...,p\} \setminus \mathcal{A}} F(\mathcal{A}, \mathcal{A} \cup \{j\})$;
    Calculate c, where c is the $1 - \alpha$ quantile of $F(1, n - p)$ with p being the size of $\mathcal{A} \cup \{j\}$;
    if $F(\mathcal{A}, \mathcal{A} \cup \{l\}) > c$ **then**
        $\mathcal{A} \leftarrow \mathcal{A} \cup \{l\}$;
    else
        Select the model with active set $\mathcal{A}$ and terminate the procedure ;
    end
end

Stepwise Regression in Linear Regression

Algorithm 5: Stepwise Regression Using F Test

Specify a significance level α ;
Specify the active set $\mathcal{A} = \emptyset$ that contains no feature ;
while *Proceed* **do**
    For each $j \in \{1, ..., p\} \setminus \mathcal{A}$, calculuate $F(\mathcal{A}, \mathcal{A} \cup \{j\})$;
    Select l such that $F(\mathcal{A}, \mathcal{A} \cup \{l\}) = \max_{j \in \{1,...,p\} \setminus \mathcal{A}} F(\mathcal{A}, \mathcal{A} \cup \{j\})$;
    Calculate c, where c is the $1 - \alpha$ quantile of $F(1, n - p)$ with p being the size of $\mathcal{A} \cup \{j\}$;
    if $F(\mathcal{A}, \mathcal{A} \cup \{l\}) > c$ **then**
        For each $j \in \mathcal{A}$, test whether it can be deleted from $\mathcal{A} \cup \{l\}$ in a similar manner to backward elimination. Update $\mathcal{A}$ to the resulting set ;
    end
    if *No feature has been added nor deleted* **then**
        Terminate the procedure ;
    end
end

Multiple Testing

A major limitation of the above procedures is that we will perform multiple F tests.

- Each F test is subject to error with probability α, where α is small.
- The aggregated error of the whole procedure is often very large or even close to 1.

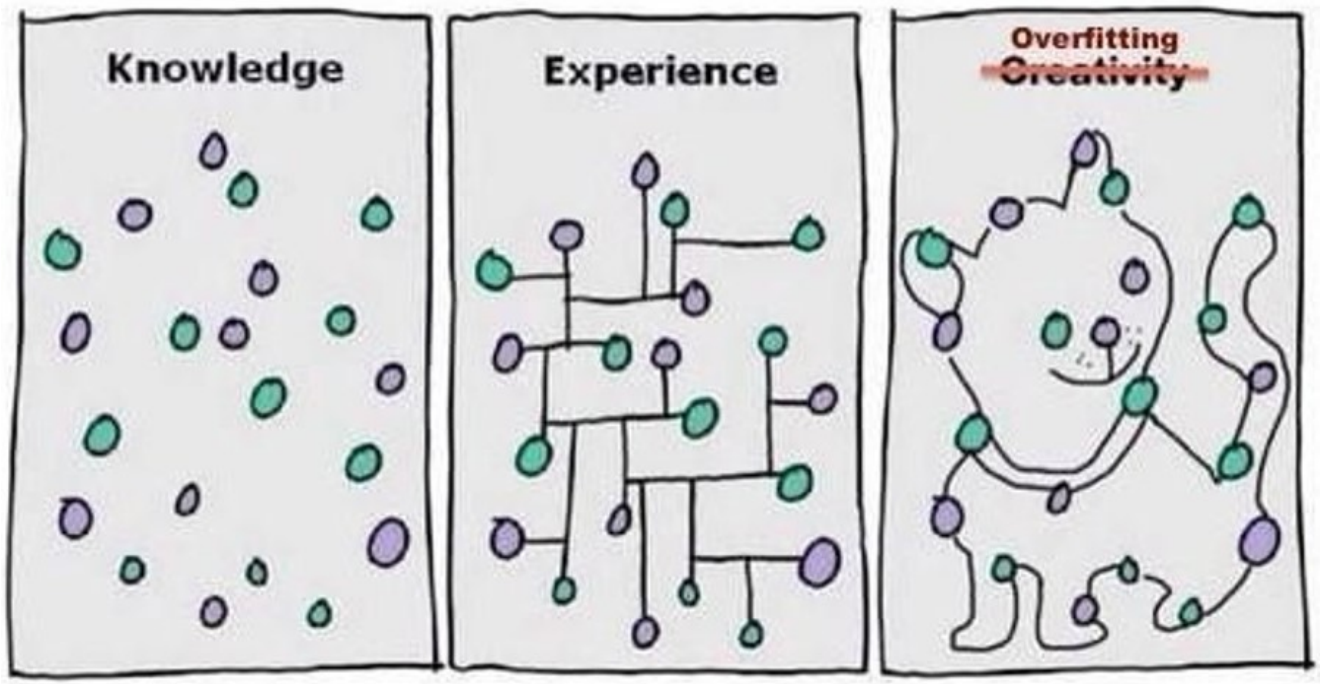
In R, backward elimination, forward selection, and stepwise regression are performed not based on the F values.

- Forward selection starts with an intercept model and sequentially adds the term that gives the lowest AIC/BIC.
- Backward elimination starts with a model with all explanatory variables and sequentially removes the term that has the greatest decrease in AIC/BIC.
- Stepwise regression is a combination of both by checking whether adding one term or deleting one term will yield the smallest AIC/BIC.

Overfitting

Suppose that $\boldsymbol{X}$ follows some distribution with density $p(\boldsymbol{X})$.

- We want our model to be complex enough to fit the data well, such as maximizing $\log p(\boldsymbol{X})$.
- On the other hand, we don't want the model to be too complex to avoid overfitting.



Information Criterion

We cannot simply choose a complex model in order to have a better fit to the current data.

- Suppose that the true data generating process is $p(x)$, but we assume $g(x \mid \theta)$, where p and g may not be the same.
- An information criterion is often of the form

$$-c \log g(\boldsymbol{X}) + \text{penality of model complexity}$$

for some constant c.

AIC: Minimizing Distance of the Fit from the Truth

We can express the KL divergence between the fitted model and the truth is

$$\mathrm{KL}(p, g) = \mathrm{E}[\log p(x^*)] - \underbrace{\int \log\left[g\left(x^* \mid \hat{\theta}\right)\right] p(x^*)\, dx^*}_{\text{define to be } R_n},$$

where we can view x^* as a new observation independent of $\boldsymbol{X}$, and $\hat{\theta} = \hat{\theta}(\boldsymbol{X})$.

- The Akaike information criterion (AIC) aims to estimate the expected value of $-R_n$. Hence, we often say AIC is a nearly unbiased estimator of KL divergence (bar some constant).
- We prefer the model with a small AIC, since a small AIC means that the divergence is small.

Derivation of AIC

Take iid data as an example, it can be shown that

$$\mathrm{E}\left[R_n\right] = \frac{1}{n}\sum_{i=1}^{n}\log g\left(x_i \mid \hat{\theta}\right) - \frac{1}{n}\mathrm{tr}\left(\mathcal{I}H^{-1}\right) + o_P\left(n^{-1}\right),$$

where

$$\mathcal{I} = \mathrm{Var}\left(\frac{\partial \log g\left(X^* \mid \theta^*\right)}{\partial\theta}\right), \text{ Fisher information}$$

$$H = -\mathrm{E}\left[\frac{\partial^2 \log g\left(x^*, \theta^*\right)}{\partial\theta\partial\theta^T}\right]. \text{ expected Hessian}$$

- If we estimate $\mathrm{tr}\left(\mathcal{I}H^{-1}\right)$ by q, the dimension of θ, we obtain the Akaike information criterion (AIC) is defined as

$$\mathrm{AIC} = -2\log\left[g\left(\boldsymbol{X}\right)\right] + 2q.$$

- Other ways of estimating $\mathrm{tr}\left(\mathcal{I}H^{-1}\right)$ is also available, yielding other information criterion such as Takeuchi information criterion (TIC).

OBS! AIC Expression

Many resources define AIC to be

$$n \log\left(\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2\right) + 2q,$$

where q is the number of parameters in $\mathrm{E}[Y]$. Keep in mind that this expression requires the normality assumption!

Under the assumption $Y_i \sim N\left(\mu_i(\theta), \sigma^2\right)$, the AIC becomes

$$\mathrm{AIC} = n \log\left(\frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{n}\right) + 2(q+1) + n\log(2\pi) + n.$$

Bayesian Information Criterion

- Bayesian information criterion penalizes a complex model much more than AIC.

$$\text{BIC} \quad = \quad -2\log\left[g\left(\boldsymbol{X}\right)\right] + \log\left(n\right)q.$$

- We prefer the model with the smallest BIC or a parsimonious model that has BIC near the minimum.
- BIC is an approximation to the so called Bayes factor in Bayesian statistics.

BIC: Consistent Model Selection

We call a model selection procedure is consistent if, with probability approaching 1, it picks the most parsimonious model among the set of models that minimize the Kullback-Leibler divergence to the truth.

- BIC is consistent.
- In contrast, AIC is not consistent and tends to be conservative.
  - The price is that, for BIC, the square risk satisfies

$$\sup_f \sum_{i=1}^{n} \mathrm{E}\left[\left(f(x_i) - \hat{f}(x_i)\right)^2\right] \to \infty, \qquad n \to \infty.$$

  - It is bounded for AIC.
- AIC is only weakly consistent.

Cross Validation (CV)

CV focuses on the prediction property.

Algorithm 6: One version of cross validation

Randomly split the data set into K nonoverlapping groups (K-fold CV) or split the data set into n groups (leave-one-out CV, aka jackknife);
for $k = 1$ *in* $1 : K$ **do**
    Take the kth group as test set and the remaining groups as training set ;
    while *for each model* **do**
        Fit it on the training set and evaluate it on the test set ;
        Retain the model performance (e.g., MSE, misclassification error, log-likelihood) ;
    end
end
Summarize the model performance (e.g., average across K groups) ;
Choose the model that performs the best ;
Refit the chosen model using the entire data set ;

Maybe a Less Discussed Point

Take regression as an example. Suppose that we have fitted a model $\hat{\mu}(x)$ and want to evaluate its prediction property.

- The test error between a new y^* and the prediction $\hat{\mu}(x^*)$ is $L(y^*, \hat{\mu}(x^*))$.
- The expected test error is $\mathrm{E}[L(y^*, \hat{\mu}(x^*)) \mid \hat{\mu}]$.
- The K-fold CV estimate of the expected test error is

$$\mathrm{CV}_K = \frac{1}{n} \sum_{k=1}^{K} \sum_{i \in C_k} L(y_i, \hat{\mu}_{-k}(x_i)),$$

  where C_k collects the indices of observations in the kth fold, and $\hat{\mu}_{-k}$ is the estimated regression function excluding the kth fold.

Such CV is a biased estimator of the expected test error!

Biased CV

In general cases,

$$\mathrm{E}\left[\mathrm{CV}_K - \mathrm{E}\left[L\left(y^*, \hat{\mu}\left(x^*\right)\right) \mid \hat{\mu}\right]\right] \approx \frac{c_0}{n\left(K-1\right)},$$

where c_0 depends only on $L\left(\right)$ and the data generating process.

- For simplicity, we consider a simple linear regression

$$Y = \beta_0 + \beta_1 X + \epsilon$$

  where $\epsilon \sim N\left(0,1\right)$ and $X \sim N\left(0, \sigma^2\right)$. We also assume $n = Km$.
- Then,

$$\mathrm{E}\left[\mathrm{CV}_K - \mathrm{E}\left[L\left(y^*, \hat{\mu}\left(x^*\right)\right) \mid \hat{\mu}\right]\right] \approx \frac{2\sigma^2}{n\left(K-1\right)},$$

$$\mathrm{Var}\left[\mathrm{CV}_K - \mathrm{E}\left[L\left(y^*, \hat{\mu}\left(x^*\right)\right) \mid \hat{\mu}\right]\right] \approx 2\sigma^4\left[\frac{1}{n} + \frac{8}{n^2} + \frac{8}{n^2\left(K-1\right)} + \frac{4}{n^2\left(K-1\right)^2} + \frac{2}{n^2\left(K-1\right)^3}\right].$$

More on Cross Validation

Suppose that the true model is one of our candidate models.

- It has been proved that, for leave-one-out CV,

$$\begin{aligned} \lim_{n\to\infty} \mathrm{P}\left(\text{Choose a model where a nonzero } \beta \text{ is missing}\right) &= 0, \\ \lim_{n\to\infty} \mathrm{P}\left(\text{Choose the most parsimonious true model}\right) &\neq 1. \end{aligned}$$

  Hence, the leave-one-out CV is conservative that selects a model of excessive size.

- It has been proved that you need leave-n_v-out CV if you want

$$\lim_{n\to\infty} \mathrm{P}\left(\text{Choose the most parsimonious true model}\right) = 1,$$

  where $n_v/n \to 1$ as $n \to \infty$.

More on Leave-One-Out Cross Validation

Suppose now that all candidate models are wrong. Consider the squared loss in linear regression

$$L_n \quad = \quad n^{-1} \left(\mathrm{E} \left(\boldsymbol{y} \mid \boldsymbol{X} \right) - \hat{\boldsymbol{y}} \right)^T \left(\mathrm{E} \left(\boldsymbol{y} \mid \boldsymbol{X} \right) - \hat{\boldsymbol{y}} \right).$$

It has been proved that, for any $\epsilon > 0$, we have

$$\lim_{n \to \infty} \mathrm{P} \left(\left| \frac{L_n \text{ of the model selected by CV}}{\text{Minimum } L_n \text{ among all candidate models}} - 1 \right| < \epsilon \right) \quad = \quad 1.$$

Cross Validation and AIC

In fact, we can show that leave-one-out CV is asymptotically equivalent to AIC if the model performance is measured by

$$\sum_{i=1}^{n} \log g\left(x_i \mid \hat{\theta}_{-i}\right),$$

where $\hat{\theta}_{-i}$ is the estimator of θ after removing the ith observation.