Cross-validation (K-fold):

Whole data set D :

C_1	C_2	C_3	$\cdots$	C_K
n_1	n_2	n_3		n_K

C_i : labels of our data points corresponding to the ith subset

$|C_i|$ = "cardinality of set C_i" = # of elements in C_i ; n_i

$D = \bigcup_{i=1}^{K} C_i$; $C_i \cap C_j = \emptyset$ (subsets are disjoint).

$T_i = D \setminus C_i = \bigcup_{\substack{j=1 \\ j \neq i}}^{K} C_j$; T_i is ith training set

$\uparrow$ "set difference" $\quad V_i = C_i$: the ith validation set

① For $i = 1, \dots, K$

(a) "train"/ fit our model on T_i , validate it on V_i.

(b) get a "discrepancy" $\text{Discr}(\hat{\underset{\sim}{Y}}^{(i)}, \underset{\sim}{Y}^{(i)})$ between predicted values $\hat{\underset{\sim}{Y}}^{(i)}$ (based on T_i) for $\underset{\sim}{Y}^{(i)}$ (from V_i).

"Discr": MSE or misclassification rate.

② Aggregate the discrepancies from $i = 1, \dots, K$ into a single measure.

How to calibrate/estimate tuning parameters using K-fold CV?

Examples:

1) polynomial regression: tuning par. is degree of polynomial.

2) GAM, ridge regression, lasso: need to choose the penalty parameter λ (for "regularization").

3) model selection: need to pick a subset of covariates.

In practice, the "aggregated discrepancy" (AD) depends on the tuning parameters (τ).

$\Rightarrow$ optimize/minimize AD with respect to τ.

Specifically, compute AD on a grid of tau values; then pick the value of tau that minimizes AD.

SML 2018 Lec. 10

Cross-Validation for Classification Problems

- We divide the data into K roughly equal-sized parts $C_1, C_2, \ldots C_K$. C_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 $n_k = n/K$.

  ✱ 1. assume $n_k = \text{const}$, so $n_k/n = 1/K$

- Compute

$$\mathrm{CV}_K = \sum_{k=1}^{K} \frac{n_k}{n} \underbrace{\mathrm{Err}_k}_{z_k}$$

$$CV_k = \bar{z} = \frac{1}{K}\sum_{j=1}^{K} z_j$$

  where $\mathrm{Err}_k = \sum_{i \in C_k} I(y_i \neq \hat{y}_i)/n_k$.

- ✱ The estimated standard deviation of $\cancel{\mathrm{CV}}_K$ Err_k is

$$\widehat{\mathrm{SE}}(\cancel{\mathrm{CV}}_K \ \mathrm{Err}_k) = \sqrt{\sum_{k=1}^{K} (\mathrm{Err}_k - \overline{\mathrm{Err}_k})^2/(K-1)}$$

- This is a useful estimate, but strictly speaking, not quite valid.

1. To estimate $Var(Z_j)$, we can use sample variance.

$Var(\bar{Z})$

2. To estimate, note $Var(\bar{Z}) = Var\left(\frac{1}{K}\sum_{j=1}^{K} Z_j\right) = \frac{1}{K^2} Var\left(\sum_{j=1}^{K} Z_j\right)$

$$= \frac{1}{K^2} \underbrace{\sum_{i=1}^{K}\sum_{j=1}^{K} Cov(Z_i, Z_j)}$$

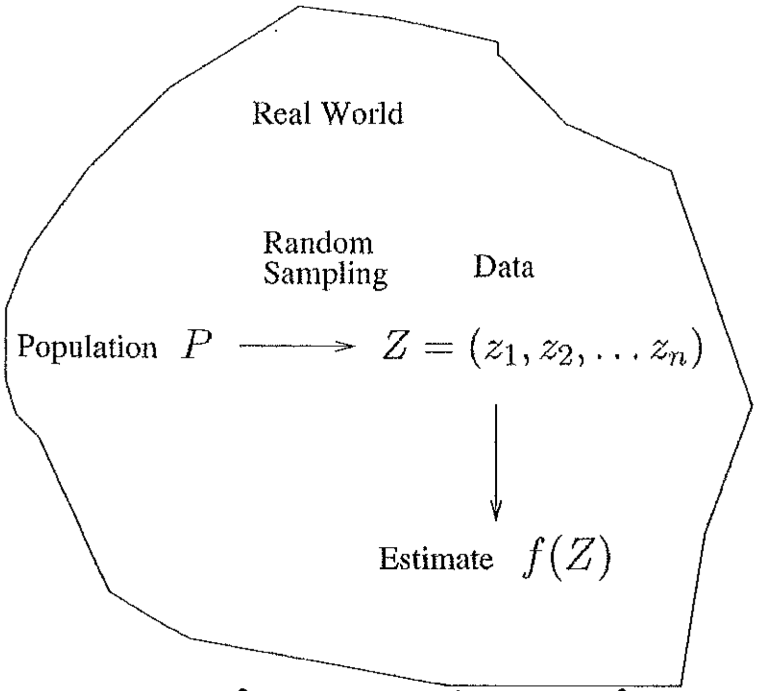
$$\sum_{j=1}^{K} Var(Z_j) + \underbrace{\sum_{i \neq j}\sum Cov(Z_i, Z_j)}_{\text{cannot be neglected for } SE(CV_K)}$$

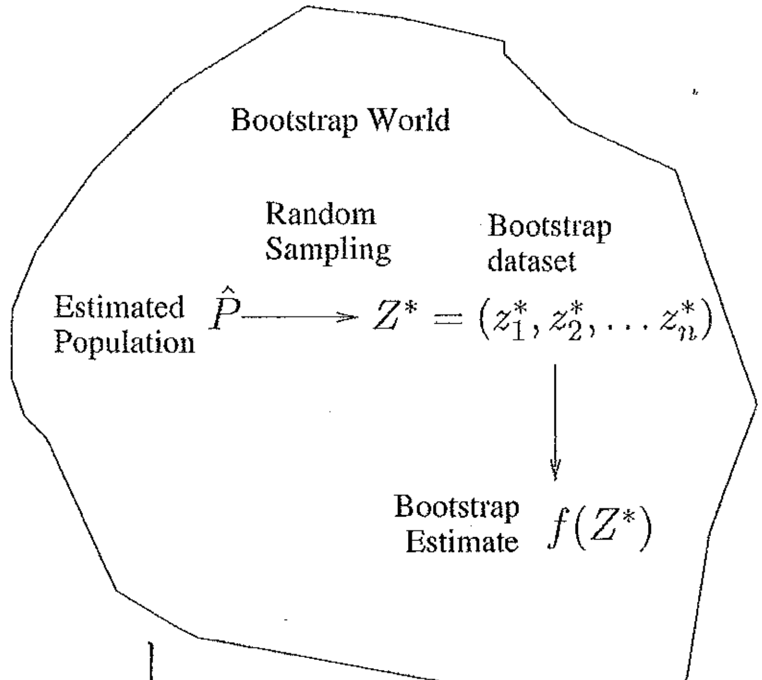
want to est. $h(P)$, estimator $f(Z)$
want its sampling distn

A general picture for the bootstrap

Case 1: finite populations.
$P = \{a_1, a_2, \dots, a_N\}$.

$\hat{P} = \{z_1, \dots, z_n\}$: our dataset





X: a rvs for sampling with replacement.

$\Pr(X = a_i) = 1/N$

$$\Pr(X \le x) = \sum_{a_i \le x} \underbrace{\Pr(X = a_i)}^{1/N} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(a_i \le x)$$

indicator fun

If n is large, $\hat{P}$ is "representative" of P.

X^*: rv for sampling with replacement

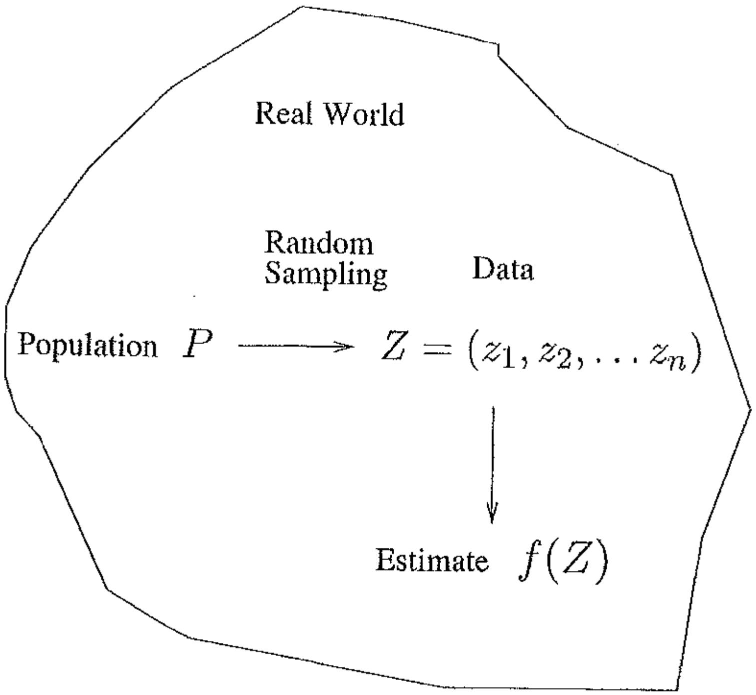
$\Pr(X^* = z_i) = 1/n$

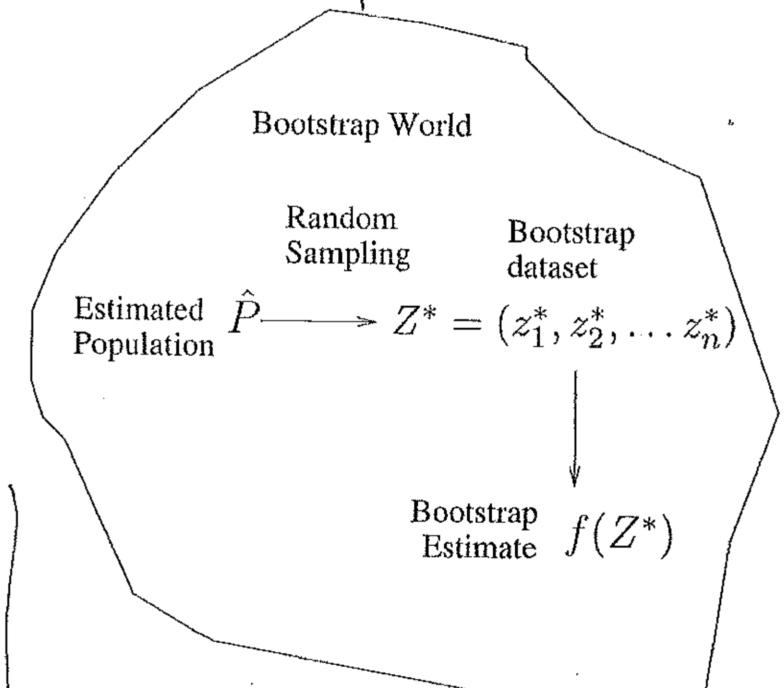
$$\Pr(X^* \le x) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}(z_i \le x)$$

A general picture for the bootstrap

Case 2: infinite populations: Think in terms of distribution functions

"non parametric bootstrap".





P is specified by a distribution function: $F(x) = Pr(X \le x)$

true cdf of the X

F(x) = Pr(X <= x)

$$\hat{F}_n(x) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(z_i \le x)$$

empirical distribution function.

$\hat{F}_n(x) \to F(x)$ as $n \to \infty$,

In R, ecdf function.

Discussed on 15-Oct-2020

Parametric Bootstrap:

Suppose we knew that the true cdf F is Normal(μ, σ^2) ; μ, σ^2 : both unknown.

How can we estimate $F(x)$ or F (the whole function)?

Plug-in estimator : estimate μ by $\bar{X}_n$, sample mean
σ^2 by S^2, sample var.

then Normal$(\mu = \bar{X}, \sigma^2 = S^2)$ is our estimator for F.