Goodness-of-Fit and Two-Sample Testing

Larry Wasserman
Department of Statistics and Data Science
and
Machine Learning Department
Carnegie Mellon University

The Problem (as I Understand It)

The Problem (as I Understand It)

We observe

$$Y_1, \dots, Y_n \sim p$$

where

$$p = (1 - \lambda) \underbrace{b(y)}_{\text{background}} + \lambda \underbrace{s(y)}_{\text{signal}} .$$

The Problem (as I Understand It)

We observe

$$Y_1, \ldots, Y_n \sim p$$

where

$$p = (1-\lambda) \underbrace{b(y)}_{\text{background}} + \lambda \underbrace{s(y)}_{\text{signal}}.$$

We want to test

$$H_0 : \lambda = 0$$

or, equivalently,

$$H_0 : p = b.$$

The Problem (as I Understand It)

We observe

$$Y_1, \ldots, Y_n \sim p$$

where

$$p = (1-\lambda) \underbrace{b(y)}_{\text{background}} + \lambda \underbrace{s(y)}_{\text{signal}}.$$

We want to test

$$H_0 : \lambda = 0$$

or, equivalently,

$$H_0 : p = b.$$

And we have many flavors such as ...

Flavors

Flavors

Background:

Flavors

Background:

- given: goodness-of-fit

Flavors

Background:

- given: goodness-of-fit
- model: goodness-of-fit with nuisance parameters

Flavors

Background:

- given: goodness-of-fit
- model: goodness-of-fit with nuisance parameters
- sampled: two-sample

Flavors

Background:

- given: goodness-of-fit
- model: goodness-of-fit with nuisance parameters
- sampled: two-sample

When a signal model is given, this becomes model dependent search.

Flavors

Background:

- given: goodness-of-fit
- model: goodness-of-fit with nuisance parameters
- sampled: two-sample

When a signal model is given, this becomes model dependent search.

My goal: pointers to the statistics literature that might be useful.

Assumed Background $b(y)$

If the background density $b(y)$ is assumed, this is a goodness of fit test:

$$Y_1, \ldots, Y_n \sim p$$

$$H_0 : p = b \quad \text{versus} \quad H_1 : p \neq b.$$

Assumed Background $b(y)$

If the background density $b(y)$ is assumed, this is a goodness of fit test:

$$Y_1, \ldots, Y_n \sim p$$

$$H_0 : p = b \quad \text{versus} \quad H_1 : p \neq b.$$

This is the classic goodness-of-fit problem but it is multivariate.

Goodness-of-fit: Optimality

Is there an optimal test?

Goodness-of-fit: Optimality

Is there an optimal test?

Yes and No.

Goodness-of-fit: Optimality

Is there an optimal test?

Yes and No.

Yes. (Ingster and Suslina 2003, Arias-Castro and Pelletier 2018, Balakrishnan and Wasserman 2019).

Goodness-of-fit: Optimality

Is there an optimal test?

Yes and No.

Yes. (Ingster and Suslina 2003, Arias-Castro and Pelletier 2018, Balakrishnan and Wasserman 2019).

$H_0 : p = b$ versus $H_1 : d(p, b) \geq \epsilon$
$p \in \mathcal{P}$ (nonparametric class: Sobolev space or Besov space) and some distance d.

Goodness-of-fit: Optimality

Is there an optimal test?

Yes and No.

Yes. (Ingster and Suslina 2003, Arias-Castro and Pelletier 2018, Balakrishnan and Wasserman 2019).

$H_0 : p = b$ versus $H_1 : d(p, b) \geq \epsilon$
$p \in \mathcal{P}$ (nonparametric class: Sobolev space or Besov space) and some distance d.

There exists a minimax test ϕ^* maximizes minimum power. That is, it achieves

$$\sup_{\phi} \inf_{d(p,b) \geq \epsilon} P(\phi = \mathrm{reject})$$

Goodness-of-fit: Optimality

Is there an optimal test?

Yes and No.

Yes. (Ingster and Suslina 2003, Arias-Castro and Pelletier 2018, Balakrishnan and Wasserman 2019).

$H_0 : p = b$ versus $H_1 : d(p, b) \geq \epsilon$
$p \in \mathcal{P}$ (nonparametric class: Sobolev space or Besov space) and some distance d.

There exists a minimax test ϕ^* maximizes minimum power. That is, it achieves

$$\sup_{\phi} \inf_{d(p,b) \geq \epsilon} P(\phi = \mathrm{reject})$$

It's optimal but the power is not high.
Generally, the likelihood ratio test is not special!

Goodness-of-fit: Optimality

Is there an optimal test?

Goodness-of-fit: Optimality

Is there an optimal test?

No. Janssen (2000) showed that any omnibus test only has substantial power in finitely many directions.

Goodness-of-fit: Optimality

Is there an optimal test?

No. Janssen (2000) showed that any omnibus test only has substantial power in finitely many directions.

Cannot distinguish close alternatives at a distance of $n^{-1/2}$.

Goodness-of-fit: Optimality

Is there an optimal test?

No. Janssen (2000) showed that any omnibus test only has substantial power in finitely many directions.

Cannot distinguish close alternatives at a distance of $n^{-1/2}$.

Nevertheless, there are some multivariate tests that you might not know which might be useful which we now review.

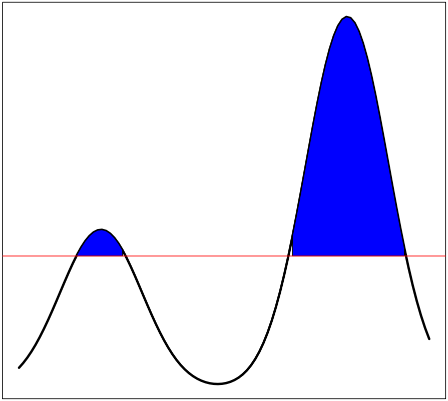
Level Set Test (Polonik 1999)

Level Set Test (Polonik 1999)

Let

$$\Gamma_t = \{y : \; b(y) \geq t\}$$

be the upper level set. This is a one-dimensional family of subsets. (VC dimension 1).



Level Set Test (Polonik 1999)

Level Set Test (Polonik 1999)

Then: $P = B$ iff $P(\Gamma_t) = B(\Gamma_t)$ for all t.

Level Set Test (Polonik 1999)

Then: $P = B$ iff $P(\Gamma_t) = B(\Gamma_t)$ for all t.

Let

$$T_n = \sup_t |P_n(\Gamma_t) - B(\Gamma_t)|$$

where

$$P_n(\Gamma_t) = \frac{1}{n} \sum_i I(Y_i \in \Gamma_t)$$

$$B(\Gamma_t) = \int_{\Gamma_t} b(y) dy$$

Level Set Test (Polonik 1999)

Then: $P = B$ iff $P(\Gamma_t) = B(\Gamma_t)$ for all t.

Let

$$T_n = \sup_t |P_n(\Gamma_t) - B(\Gamma_t)|$$

where

$$P_n(\Gamma_t) = \frac{1}{n}\sum_i I(Y_i \in \Gamma_t)$$

$$B(\Gamma_t) = \int_{\Gamma_t} b(y)dy$$

Then

$$\sqrt{n}T_n \rightsquigarrow \sup_t |\mathbb{G}(t)|$$

where $\mathbb{G}$ is a Gaussian process. This is distribution free. Like a KS test.

Bickel-Breiman Nearest Neighbor Test

Bickel-Breiman Nearest Neighbor Test

Let

$$W_i = e^{-nb(Y_i)V_i}$$

where V_i is the volume of the ball containing the nearest neighbor.

Bickel-Breiman Nearest Neighbor Test

Let

$$W_i = e^{-nb(Y_i)V_i}$$

where V_i is the volume of the ball containing the nearest neighbor.

Let

$$F_n(t) = \frac{1}{n}\sum_i I(W_i \leq t)$$

and

$$S = \int F_n^2(t)dt$$

which has a known limiting distribution. See Schilling (1983).

Neyman Smooth Test

This test targets certain directions by specifiying basis functions $\phi_1, \phi_2, \ldots$.

Neyman Smooth Test

This test targets certain directions by specifiying basis functions $\phi_1, \phi_2, \ldots$.

Model $p(y)$ as

$$p(y) = b(y)e^{\sum_j \theta_j \phi_j(y) - Z}.$$

Neyman Smooth Test

This test targets certain directions by specifiying basis functions $\phi_1, \phi_2, \ldots$.

Model $p(y)$ as

$$p(y) = b(y) e^{\sum_j \theta_j \phi_j(y) - Z}.$$

Easy to estimate the θ_j's and then test $\theta = 0$.

Neyman Smooth Test

This test targets certain directions by specifiying basis functions $\phi_1, \phi_2, \ldots$.

Model $p(y)$ as

$$p(y) = b(y)e^{\sum_j \theta_j \phi_j(y) - Z}.$$

Easy to estimate the θ_j's and then test $\theta = 0$.

See Algeri (2020, 2021).

With a Given Signal (Model Dependent)

For a given signal s, the LRT is

$$T = \sup_{\lambda} \prod_i \left(1 - \lambda + \lambda \frac{s(Y_i)}{b(Y_i)}\right)$$

With a Given Signal (Model Dependent)

For a given signal s, the LRT is

$$T = \sup_{\lambda} \prod_{i} \left(1 - \lambda + \lambda \frac{s(Y_i)}{b(Y_i)} \right)$$

A possibly better test is the score test:

With a Given Signal (Model Dependent)

For a given signal s, the LRT is

$$T = \sup_{\lambda} \prod_i \left(1 - \lambda + \lambda \frac{s(Y_i)}{b(Y_i)}\right)$$

A possibly better test is the score test:

$$T = \frac{1}{n} \sum_i \frac{s(Y_i)}{b(Y_i)} - 1$$

which does not require estimating λ

With a Signal Model

This is a parametric family:

$$p(y) = (1 - \lambda)b(y) + \lambda s_\theta(y)$$

(or perhaps for a one-dimensional marginal such as mass).

With a Signal Model

This is a parametric family:

$$p(y) = (1 - \lambda)b(y) + \lambda s_\theta(y)$$

(or perhaps for a one-dimensional marginal such as mass).

λ and θ can be estimated by maximum likelihood using the EM algorithm.

With a Signal Model

This is a parametric family:

$$p(y) = (1 - \lambda)b(y) + \lambda s_\theta(y)$$

(or perhaps for a one-dimensional marginal such as mass).

λ and θ can be estimated by maximum likelihood using the EM algorithm.

Testing λ is tricky because of the boundary and because θ is not identified under H_0. LRT has nonstandard limiting behavior.

With a Signal Model

This is a parametric family:

$$p(y) = (1 - \lambda)b(y) + \lambda s_\theta(y)$$

(or perhaps for a one-dimensional marginal such as mass).

λ and θ can be estimated by maximum likelihood using the EM algorithm.

Testing λ is tricky because of the boundary and because θ is not identified under H_0. LRT has nonstandard limiting behavior.

Max score:

$$T_n = \sup_\theta \frac{1}{n} \sum_i \frac{s_\theta(Y_i)}{b(Y_i)} - 1$$

and the null distribution can be obtained by simulation.

Bump Test

Bump Test

Target the bumps in a one dimensional marginal $M = f(Y)$.

Bump Test

Target the bumps in a one dimensional marginal $M = f(Y)$.

Test:
$H_0 : p(m) = b(m)$ for all m versus $H_1 : p(m) > b(m)$ for some m.

Bump Test

Target the bumps in a one dimensional marginal $M = f(Y)$.

Test:
$H_0 : p(m) = b(m)$ for all m versus $H_1 : p(m) > b(m)$ for some m.

Don't use histograms! Use the local polynomial density estimator $\widehat{p}$ (Cattaneo, Jansson and Ma 2020).

Bump Test

Target the bumps in a one dimensional marginal $M = f(Y)$.

Test:
$H_0 : p(m) = b(m)$ for all m versus $H_1 : p(m) > b(m)$ for some m.

Don't use histograms! Use the local polynomial density estimator $\widehat{p}$ (Cattaneo, Jansson and Ma 2020).

$F(m) = P(M \leq m)$.

Bump Test

Target the bumps in a one dimensional marginal $M = f(Y)$.

Test:
$H_0 : p(m) = b(m)$ for all m versus $H_1 : p(m) > b(m)$ for some m.

Don't use histograms! Use the local polynomial density estimator $\widehat{p}$ (Cattaneo, Jansson and Ma 2020).

$F(m) = P(M \leq m)$.

For u near m:

$$\begin{aligned} F(u) &\approx F(m) + (u-m)p(m) + \frac{(u-m)^2}{2}p'(m) \\ &= \beta_0(m) + (u-m)\beta_1(m) + (u-m)^2\beta_2(m) \end{aligned}$$

Bump Test

Bump Test

Let

$$F_n(m) = \frac{1}{n} \sum_i I(M_i \leq m)$$

Bump Test

Let

$$F_n(m) = \frac{1}{n} \sum_i I(M_i \leq m)$$

Let $\widehat{\beta}(m)$ minimize:

$$\min_b \sum_i (\widehat{F}_n(m) - r^T b)^2 \; K\left(\frac{M_i - m}{h(m)}\right)$$

where $r = (1, M_i - m, (M_i - m)^2)$, K is a kernel, $h(m) = (C(m)/n)^{1/5}$ and $C(m)$ is known.

Bump Test

Let

$$F_n(m) = \frac{1}{n} \sum_i I(M_i \leq m)$$

Let $\widehat{\beta}(m)$ minimize:

$$\min_b \sum_i (\widehat{F}_n(m) - r^T b)^2 \; K\left(\frac{M_i - m}{h(m)}\right)$$

where $r = (1, M_i - m, (M_i - m)^2)$, K is a kernel, $h(m) = (C(m)/n)^{1/5}$ and $C(m)$ is known.

Let

$$\widehat{p}(m) = \widehat{\beta}_1(m).$$

This is optimal (under mild conditions) and boundary adaptive.

Bump Test

Let

$$F_n(m) = \frac{1}{n} \sum_i I(M_i \le m)$$

Let $\widehat{\beta}(m)$ minimize:

$$\min_b \sum_i (\widehat{F}_n(m) - r^T b)^2 \; K\left(\frac{M_i - m}{h(m)}\right)$$

where $r = (1, M_i - m, (M_i - m)^2)$, K is a kernel,
$h(m) = (C(m)/n)^{1/5}$ and $C(m)$ is known.

Let

$$\widehat{p}(m) = \widehat{\beta}_1(m).$$

This is optimal (under mild conditions) and boundary adaptive.

Use

$$T = \sup_m [\widehat{p}(m) - b(m)].$$

Robustness to Background Misspecification

Robustness to Background Misspecification

There is a growing literature on robust tests:

$$H_0 : Y_1, \ldots, Y_n \sim q, \;\; q \in N_\epsilon(b)$$

where $N_\epsilon(b)$ is a neighborhood of b.

Robustness to Background Misspecification

There is a growing literature on robust tests:

$$H_0 : Y_1, \ldots, Y_n \sim q, \;\; q \in N_\epsilon(b)$$

where $N_\epsilon(b)$ is a neighborhood of b.

Examples: Wasserstein neighborhood (Xie, Gao and Xie 2021) RKHS neighborhood (Sun and Zou 2022), Huber neighborhood (Huber 1965).

Robustness to Background Misspecification

There is a growing literature on robust tests:

$$H_0 : Y_1, \dots, Y_n \sim q, \;\; q \in N_\epsilon(b)$$

where $N_\epsilon(b)$ is a neighborhood of b.

Examples: Wasserstein neighborhood (Xie, Gao and Xie 2021) RKHS neighborhood (Sun and Zou 2022), Huber neighborhood (Huber 1965).

Tradeoff between robustness and power.

With Simulated Background: Two Sample Test

$$
\begin{aligned}
X_1, \ldots, X_m &\sim b \\
Y_1, \ldots, Y_n &\sim p = (1-\lambda)b + \lambda s
\end{aligned}
$$

With Simulated Background: Two Sample Test

$$X_1, \ldots, X_m \sim b$$
$$Y_1, \ldots, Y_n \sim p = (1-\lambda)b + \lambda s$$

Two sample test:

$$H_0 : p = b \quad \text{versus} \quad p \neq b$$

With Simulated Background: Two Sample Test

$$X_1, \ldots, X_m \sim b$$
$$Y_1, \ldots, Y_n \sim p = (1-\lambda)b + \lambda s$$

Two sample test:

$$H_0 : p = b \quad \text{versus} \quad p \neq b$$

Again there are many tests. There is no optimal test.

RKHS, MMD, Energy

RKHS, MMD, Energy

Let

$$\psi = \sup_{f \in \mathcal{F}} \Big| \mathbb{E}_B[f(X)] - \mathbb{E}_P[f(Y)] \Big|$$

where $\mathcal{F}$ is a reproducing kernel Hilbert space (RKHS).

RKHS, MMD, Energy

Let

$$\psi = \sup_{f \in \mathcal{F}} \Big| \mathbb{E}_B[f(X)] - \mathbb{E}_P[f(Y)] \Big|$$

where $\mathcal{F}$ is a reproducing kernel Hilbert space (RKHS).

$$T_n = \frac{1}{n(n-1)} \sum_{i \neq j} H_{ij}$$

where

$$H_{ij} = K(X_i, X_j) + K(Y_i, Y_j) - K(X_i, Y_j) - K(X_j, Y_i).$$

RKHS, MMD, Energy

Let

$$\psi = \sup_{f \in \mathcal{F}} \left| \mathbb{E}_B[f(X)] - \mathbb{E}_P[f(Y)] \right|$$

where $\mathcal{F}$ is a reproducing kernel Hilbert space (RKHS).

$$T_n = \frac{1}{n(n-1)} \sum_{i \neq j} H_{ij}$$

where

$$H_{ij} = K(X_i, X_j) + K(Y_i, Y_j) - K(X_i, Y_j) - K(X_j, Y_i).$$

Null distribution is complicated.

Classifier Tests

S	1	1	$\cdots$	1	0	0	$\cdots$	0
Z	Y_1	Y_2	$\cdots$	Y_m	X_1	X_2	$\cdots$	X_n

$$\begin{aligned} h(z) &= P(S=1|Z=z) \\ &= \frac{p(z|S=1)P(S=1)}{p(z|S=1)P(S=1)+p(z|S=0)P(S=0)} \\ &= \frac{p(z|S=1)\pi}{p(z|S=1)\pi+p(z|S=0)(1-\pi)} \end{aligned}$$

where $\pi = m/(m+n)$. Hence

$$\frac{p(y)}{b(y)} = \frac{h(z)}{1-h(z)}$$

so we have estimated the density ratio.
Chakravarti, Kuusela, Lei and Wasserman 2022

Classifier Tests

Classifier Tests

Which classifier?

Classifier Tests

Which classifier?

Current fashion: neural nets (deep learning)

Classifier Tests

Which classifier?

Current fashion: neural nets (deep learning)

Others:
random forests, logistic regression, ...

Classifier Tests

Which classifier?

Current fashion: neural nets (deep learning)

Others:
random forests, logistic regression, ...

Aside: why did everyone start calling classification and regression Machine Learning? It's statistics! We've been doing it for 100 years!

Classifier Tests

Which test?

Classifier Tests

Which test?

$$\prod_i \frac{\widehat{h}(Z_i)}{1-\widehat{h}(Z_i)}$$

is an estimate of the Neyman-Pearson test.

Classifier Tests

Which test?

$$\prod_i \frac{\widehat{h}(Z_i)}{1-\widehat{h}(Z_i)}$$

is an estimate of the Neyman-Pearson test.

Really, the classifier is just a dimension reduction method. We have

$$\widehat{h}(X_1),\ldots,\widehat{h}(X_N)$$

and

$$\widehat{h}(Y_1),\ldots,\widehat{h}(Y_n)$$

Classifier Tests

Which test?

$$\prod_i \frac{\widehat{h}(Z_i)}{1 - \widehat{h}(Z_i)}$$

is an estimate of the Neyman-Pearson test.

Really, the classifier is just a dimension reduction method. We have

$$\widehat{h}(X_1), \ldots, \widehat{h}(X_N)$$

and

$$\widehat{h}(Y_1), \ldots, \widehat{h}(Y_n)$$

The data are now one-dimensional. We can use any one-dimensional two-sample test we want.

Classifier Tests

Which test?

$$\prod_i \frac{\widehat{h}(Z_i)}{1 - \widehat{h}(Z_i)}$$

is an estimate of the Neyman-Pearson test.

Really, the classifier is just a dimension reduction method. We have

$$\widehat{h}(X_1), \ldots, \widehat{h}(X_N)$$

and

$$\widehat{h}(Y_1), \ldots, \widehat{h}(Y_n)$$

The data are now one-dimensional. We can use any one-dimensional two-sample test we want.

For example:
classifier accuracy, density ratio (Neyman-Pearson), KS test, etc.

Error Control

Constructing the classifier and doing the test on the same data can lead to invalid p-value.

Error Control

Constructing the classifier and doing the test on the same data can lead to invalid p-value.

Two fixes: permuations and data splitting

Error Control

Constructing the classifier and doing the test on the same data can lead to invalid p-value.

Two fixes: permuations and data splitting

Permutation: permute the labels, repeat the classifier K times, and the p-value is

$$\frac{1}{K}\sum_{j} I(T_j > t)$$

is a valid p-value. But this is expensive.

Error Control

Constructing the classifier and doing the test on the same data can lead to invalid p-value.

Two fixes: permuations and data splitting

Permutation: permute the labels, repeat the classifier K times, and the p-value is

$$\frac{1}{K}\sum_j I(T_j > t)$$

is a valid p-value. But this is expensive.

Or: split the sample. Construct the classifier on first half. Conduct the test on the second half.
See Chakravarti, Kuusela, Lei and Wasserman (2022).

Classifier Tests

Are classifier tests better than other tests?

Classifier Tests

Are classifier tests better than other tests?

No one knows.

Classifier Tests

Are classifier tests better than other tests?

No one knows.

The theoretical properties of black box classifiers (random forests, neural nets) are not understood.

Classifier Tests

Are classifier tests better than other tests?

No one knows.

The theoretical properties of black box classifiers (random forests, neural nets) are not understood.

Don't assume that neural nets are optimal.

Conclusion

Conclusion

Because there is no optimal test, we need to choose a test carefully.

Conclusion

Because there is no optimal test, we need to choose a test carefully.

Classifier tests seem very promising.

Conclusion

Because there is no optimal test, we need to choose a test carefully.

Classifier tests seem very promising.

We have virtually no theory for these tests. (Some limited results in Kim, Ramdas, Singh and Wasserman 2021).

Conclusion

Because there is no optimal test, we need to choose a test carefully.

Classifier tests seem very promising.

We have virtually no theory for these tests. (Some limited results in Kim, Ramdas, Singh and Wasserman 2021).

THE END

Error Control: Universal Inference

Error Control: Universal Inference

Exact inference, no regularity conditions. (Wasserman, Ramdas, Balakrishnan 2020).

Error Control: Universal Inference

Exact inference, no regularity conditions. (Wasserman, Ramdas, Balakrishnan 2020).

Split data: $\mathcal{D}_0$ and $\mathcal{D}_1$.

Error Control: Universal Inference

Exact inference, no regularity conditions. (Wasserman, Ramdas, Balakrishnan 2020).

Split data: $\mathcal{D}_0$ and $\mathcal{D}_1$.

Compute $\widehat{\lambda}$ from $\mathcal{D}_1$ and likelihood $\mathcal{L}_0$ from $\mathcal{D}_0$.

Error Control: Universal Inference

Exact inference, no regularity conditions. (Wasserman, Ramdas, Balakrishnan 2020).

Split data: $\mathcal{D}_0$ and $\mathcal{D}_1$.

Compute $\widehat{\lambda}$ from $\mathcal{D}_1$ and likelihood $\mathcal{L}_0$ from $\mathcal{D}_0$.

Let

$$U = \frac{\mathcal{L}_0(\widehat{\lambda})}{\mathcal{L}(0)}.$$

Error Control: Universal Inference

Exact inference, no regularity conditions. (Wasserman, Ramdas, Balakrishnan 2020).

Split data: $\mathcal{D}_0$ and $\mathcal{D}_1$.

Compute $\widehat{\lambda}$ from $\mathcal{D}_1$ and likelihood $\mathcal{L}_0$ from $\mathcal{D}_0$.

Let

$$U = \frac{\mathcal{L}_0(\widehat{\lambda})}{\mathcal{L}(0)}.$$

Repeat B times and let $U = B^{-1}\sum_j U_j$.

Error Control: Universal Inference

Exact inference, no regularity conditions. (Wasserman, Ramdas, Balakrishnan 2020).

Split data: $\mathcal{D}_0$ and $\mathcal{D}_1$.

Compute $\widehat{\lambda}$ from $\mathcal{D}_1$ and likelihood $\mathcal{L}_0$ from $\mathcal{D}_0$.

Let

$$U = \frac{\mathcal{L}_0(\widehat{\lambda})}{\mathcal{L}(0)}.$$

Repeat B times and let $U = B^{-1}\sum_j U_j$.

Reject of $U > 1/\alpha$.