# The StepM Proceedure, Model Confidence Set and False Discovery Rate Control

The Econometrics of Predictability

*This version: May 27, 2014*

May 27, 2014

UNIVERSITY OF OXFORD

# Overview

- Multiple Hypothesis Testing
  - StepM
  - Model Confidence Set
  - False Discovery Rate Control

# Stepwise Multiple Testing

- The main issue with the Reality Check and the Test for SPA is the null
- These tests ultimately test one question:
  - Is the largest out-performance consistent with a random draw from the distribution when there are not superior models to the benchmark?
- If the null is rejected, only the best performing model can be determined to be better than the benchmark
- What about the 2nd best model? Or the k$^{th}$ best model?
- The *StepM* extends that reality check by allowing individual models to be tested
- It is implemented by repeatedly applying a RC-like algorithm which controls the *Familywise Error Rate (FWE)*

# Basic Setup

- The basic setup is identical to that of the RC/SPA
- The test is based on $\delta_{j,t} = L\left(y_{t+h}, \hat{y}_{t+h,BM|t}\right) - L\left(y_{t+h}, \hat{y}_{t+h,j|t}\right)$
- Can be used in the same types of tests as RC/SPA
  - Absolute return
  - Sharpe Ratio
  - Risk-adjusted $\alpha$ comparisons
  - MSE/MAE
  - Predictive Likelihood
- Can be implemented on both raw and Studentized loss differentials

# Null and Alternative Hypotheses

- The null and alternatives in StepM are not a single statement as they were in the RC/SPA
- The null**s** are

$$H_{0,j} : \mathrm{E}\left[\delta_t\right] \leq 0, \quad j = 1, \ldots, m$$

- The alternative**s** are

$$H_{1,j} : \mathrm{E}\left[\delta_t\right] > 0, \quad j = 1, \ldots, m$$

- StepM will ultimately result in a set of rejections (if any are rejected)
- Goal of StepM is to identify as many false nulls as possible while controlling the Familywise Error Rate

# Familywise Error Rate

## Definition (Familywise Error Rate)

For a set of null and alternative hypotheses $H_{0,i}$ and $H_{1,i}$ for $i = 1, \ldots, m$, let $\mathcal{I}_0$ contain the indices of the correct null hypotheses. The Familywise Error Rate is defined as

$$\Pr\left(\text{Rejecting at least one } H_{0,i} \text{ for } i \in \mathcal{I}_0\right) = 1 - \Pr\left(\text{Reject no } H_{0,i} \text{ for } i \in \mathcal{I}_0\right)$$

- The FWE is concerned only with the probability of making at least one Type I error
- Making 1, 2 or $m$ Type I errors is the same to FWE
  - This is a criticism of FWE
  - Other criteria exist such as *False Discovery Rate* which controls the percentage of rejections which are false (# False Rejection/# Rejections)

# Bonferoni Bounds

- Bonferoni bounds are the first procedure to control FWE

## Definition (Bonferoni Bound)

Let $T_1, T_{2,}, \ldots, T_m$ be a set of $m$ test statistics, then

$$\underbrace{\Pr\left(T_1 \cup \ldots \cup T_m | H_{1,0}, \ldots H_{m,0}\right)}_{\text{Joint Probability}} \leq \sum_{j=1}^{m} \underbrace{\Pr\left(T_j | H_{0,j}\right)}_{\text{Individual Probability}}$$

where $\Pr\left(T_j | H_{0,j}\right)$ is the probability of observing $T_j$given the null $H_{0,j}$ is true.

- Bonferoni bounds are a simple method to test $m$ hypotheses using only univariate test statistics
- Let $\{pv_j\}$ be a set of $m$ p-values from a set of tests
- The Bonferoni bound will reject the set of nulls is $pv_j \leq \alpha/m$ for all $j$
  - $\alpha$ is the size of the test (e.g. 5%)
- When $m$ is moderately large, this is a very conservative test
- Conservative since assumes worst case dependence among statistics

# Holm’s procedure

## Definition (Holm’s Procedure)

Let $T_1, T_{2,} \ldots, T_m$ be a set of $m$ test statistics with associated p-values $pv_j$, $j = 1, \ldots, m$ where it is assumed $pv_i < pv_j$ if $i < j$. If

$$pv_j \leq \alpha / \left(m - j + 1\right)$$

then $H_{0,j}$ can be rejected in factor of $H_{1,j}$ while controlling the famliywise error rate at $\alpha$.

- Example: p-values of $.001$, $.01$, $.03$, $.05$, $m = 4$, $\alpha = .05$
- Improves Bonferoni by ordering the p-values and using a stepwise procedure
- Allows subsets of hypotheses to be tested – Bonferoni is joint
- Less strict, except when $j = 1$ (same as Bonferoni)
- **Note**: Holm’s procedure ends as soon as a null cannot be rejected

# Relationships between testing procedures

- The RC/SPA, Bonferoni and Holm are all related

| | Worst-case Dependence | Accounts for Dependence in Data |
|---|---|---|
| Single-step | Bonferoni | RC, SPA |
| Stepwise | Holm | StepM |

# StepM Algorithm

## Algorithm (StepM)

```
1. Begin with the active set $\mathcal{A} = \{1, 2, \ldots, m\}$, superior set $\mathcal{S} = \{\}$
2. Construct $B$ bootstraps sample $\left\{\boldsymbol{\delta}^{\star}_{b,t}\right\}$, $b = 1, \ldots, B$
3. For each bootstrap sample, compute $T^{\star StepM}_{k,b} = \max_{j\in\mathcal{A}} \left\{\bar{\delta}^{\star}_{b,j} - \bar{\delta}_j\right\}$
4. Compute $q_{k,\alpha}$ as the $1 - \alpha$ quantile of $\left\{T^{\star StepM}_{k,b}\right\}$
5. If $\max_{j\in\mathcal{A}} \left(\bar{\delta}_j\right) < q_{k,\alpha}$ stop
6. Otherwise for each $j \in \mathcal{A}$
    a. If $\bar{\delta}_j \geq q_{k,\alpha}$ add $j$ to $\mathcal{S}$ and delete from $\mathcal{A}$
    b. Return to 2
```

# Comments

- StepM would be virtually identical to RC if only the largest $\bar{\delta}_j$ was tested
- Improves on the RC since (weakly more) individual out-performing models can be identified
- If no model outperforms, will stop with none and RC p-value will be larger than $\alpha$
- Steps 2−4 are identical to the RC using the models in $\mathcal{A}$
- The stepwise testing can improve power by removing models
    - The improvement comes if a model with substantial out-performance also has large variance
    - Removing this model allows the critical value to be reduced
- StepM only guarantees that FWE$\leq \alpha$, and in general will be $< \alpha$
    - Will only $= \alpha$ if $\mathrm{E}\left[\delta_{j,t}\right] = 0$ for all $j$
    - Example: $N\left(\mu, \sigma^2\right)$ when $\mu < 0, H_0 : \mu = 0$

# Studentization

- Like the SPA to the RC, the StepM can be implemented using Studentized loss differentials
- Romano & Wolf argue that the Studentization should be done *inside* each bootstrap sample, not globally as in the SPA
- Theoretically both are justified and neither makes a difference asymptotically
- Computing the variance inside each bootstrap will more closely match the re-sampled data than when using a global estimate

# Studentized StepM Algorithm

## Algorithm (Studentized StepM)

1. *Begin with the active set* $\mathcal{A} = \{1, 2, \ldots, m\}$*, superior set* $\mathcal{S} = \{\}$
2. *Compute* $\bar{z}_j = \bar{\delta}_j / \sqrt{\hat{\omega}_j^2 / P}$ *where* $\hat{\omega}_j^2$ *was previously defined*
3. *Construct B bootstraps sample* $\left\{ \boldsymbol{\delta}_{b,t}^{\star} \right\}, b = 1, \ldots, B$
4. *For each bootstrap sample, compute*

$$T_{k,b}^{\star StepM} \quad = \quad \max_{j \in \mathcal{A}} \left\{ \frac{\bar{\delta}_{b,j}^{\star} - \bar{\delta}_j}{\hat{\omega}_j^{\star}} \right\}$$

*where* $\hat{\omega}_j^{2\star}$ *is an estimate of the long-run variance of the bootstrapped data*

5. *Compute* $q_{k,\alpha}^{z}$ *as the* $1 - \alpha$ *quantile of* $\left\{ T_{k,b}^{\star StepM} \right\}$
6. *If* $\max_{j \in \mathcal{A}} \left( \bar{z}_j \right) < q_{k,\alpha}^{z}$ *stop*
7. *Otherwise for each* $j \in \mathcal{A}$
   a. *If* $\bar{z}_j \geq q_{k,\alpha}^{z}$ *add j to* $\mathcal{S}$ *and delete from* $\mathcal{A}$
   b. *Return to 2*

# Why Studentization Help

- StepM is built around confidence intervals of the form

$$\left[\bar{\delta}_1 - q_{1,\alpha}, \infty\right] \times \ldots \times \left[\bar{\delta}_m - q_{1,\alpha}, \infty\right]$$

- Null hypotheses are rejected for models where 0 is *not* in its confidence interval
- In the raw form, the confidence interval is a square – the same for every loss differential
- When Studentization is used, the confidence intervals take the form

$$\left[\bar{\delta}_1 - \sqrt{\omega_1^2/P} q^z_{1,\alpha}, \infty\right] \times \ldots \times \left[\bar{\delta}_m - \sqrt{\omega_m^2/P} q^z_{1,\alpha}, \infty\right]$$

- This “customization” allows for more rejections if the loss differentials have cross-sectional heteroskedasticity

# Block-size Selection

- Paper proposes a procedure to make data driven block size
- Basic idea is to use a (V)AR on $\left\{\delta_{j,t}\right\}$ to approximate the dependence
  - Similar to Den Hann-Levine HAC
- Fit AR & estimate residual covariance (or use short block bootstrap on errors)
- Simulate from model
- For $w = 1, \ldots, \overline{W}$ compute the bootstrap confidence region with size $1 - \alpha$using percentile method
- For each block size, compute the empirical coverage – percentage of simulated $\bar{\delta}$ in their confidence region
- Choose optimal $w$ which most closely matches $1 - \alpha$
  - Alternative: Use Politis & White

# Empirical Application

- Applied StepM to a set of 105 Hedge Fund Returns with long histories
- Returns net of management fees
- Benchmark model was *risk-free rate*
- $m = 105, P = 147$ (all out-of-sample)
- Results:
  - Raw data: No out-performers
    - Max ratio of standard deviation $\hat{\omega}_i / \hat{\omega}_j = 22$
  - Studentized: 7 funds identified
- **Note**: Will *always* identify funds with the largest $\bar{\delta}$ (or $\bar{z}$) first

# Empirical Application

| $\bar{x}_{T,s} - \bar{x}_{T,S+1}$ | Fund | $(\bar{x}_{T,s} - \bar{x}_{T,S+1})/\hat{\sigma}_{T,s}$ | Fund |
|---|---|---|---|
| 1.70 | Libra Fund | 10.63 | Market Neutral* |
| 1.41 | Private Investment Fund | 9.26 | Market Neutral Arbitrage* |
| 1.36 | Aggressive Appreciation | 8.43 | Univest (B)* |
| 1.27 | Gamut Investments | 6.33 | TQA Arbitrage Fund* |
| 1.26 | Turnberry Capital | 5.48 | Event-Driven Risk Arbitrage* |
| 1.14 | FBR Weston | 5.29 | Gabelli Associates* |
| 1.11 | Berkshire Partnership | 5.24 | Elliott Associates** |
| 1.09 | Eagle Capital | 5.11 | Event Driven Median |
| 1.07 | York Capital | 4.97 | Halcyon Fund |
| 1.07 | Gabelli Intl. | 4.65 | Mesirow Arbitrage Trust |

# Improving StepM using SPA

- The main step in the StepM algorithm is identical to the RC
- The important difference is that the test is implemented for each null, rather than globally
- StepM will suffer if very poor models are included with a large variance
  - Especially true for raw version, but also relevant for Studentized version
  - Example

$$\left[\begin{array}{c}\bar{\delta}_1\\ \bar{\delta}_2\end{array}\right] \sim N\left(\left[\begin{array}{c}0\\ -5\end{array}\right], \left[\begin{array}{cc}1 & 0\\ 0 & 1\end{array}\right]\right)$$

  - Reality Check critical value will be 1.95, while “best” critical value would be 1.645 (since only 1 relevant for asymptotic distribution)
- The RC portions of StepM can be replaced by SPA versions which addresses this problem
- Simple as adding in the indicator function $I_j^c$ when subtracting the mean in step 3 (step 4 in Studentized version)
- Using SPA modification will always find more out-performing models

# Model Confidence Set (MCS)

- RC, SPA and StepM were all testing superior predictive ability
- This type hypothesis is common when there is a natural benchmark
- In some scenarios there may not be a single benchmark, or there may more than one models which could be considered benchmarks
- When this occurs, it is not clear
  - How to implement RC/SPA/StepM
  - How to make sound conclusions about superior predictive ability
- The model confidence set addresses this problem by *bypassing the benchmark*
- The MCS aims to find the *best model* and all models which are *indistinguishable from the best*
  - The model with the lowest loss will always be the best – identifying the others is more challenging
- Also returns p-values for models with respect to the MCS

# Notation Preliminaries

- The outcome of the MCS is a *set of models*
  - All model sets will be denoted using $\mathcal{M}$
- The initial model set is $\mathcal{M}_0$
- The goal is to find $\mathcal{M}^{\star}$ which is the set of all models which are indistinguishable from the best
- The output of the MCS algorithm is $\widehat{\mathcal{M}}_{1-\alpha}$ where $\alpha$ is the size of the test
  - The size is interpreted as a Familywise Error Rate – same as StepM
  - In general $\widehat{\mathcal{M}}_{1-\alpha}$ will contain more than 1 model
- In between $\mathcal{M}_0$ and $\widehat{\mathcal{M}}_{1-\alpha}$ are other sets of models

$$\mathcal{M}_0 \supset \mathcal{M}_1 \supset \ldots \supset \widehat{\mathcal{M}}_{1-\alpha}$$

# Notation Preliminaries

- To construct the model confidence set, two tools are needed
  - An equivalence test $d_{\mathcal{M}}$: Determines whether the model in $\mathcal{M}$ are equal in terms of loss
  - An elimination rule $e_{\mathcal{M}}$: Determines which model to eliminate if $d_{\mathcal{M}}$ finds that the models are not equivalent
- The generic form of the algorithm, starting at $i = 0$:
  1. Apply $d_{\mathcal{M}}$ to $\mathcal{M}_i$
  2. If $d_{\mathcal{M}}$ rejects equivalence, use $e_{\mathcal{M}}$to eliminate 1 model to produce $\mathcal{M}_{i+1}$
     a. If not, stop
  3. Increment $i$, return to 1
- Has a similar flavor to StepM
  - Also gains from eliminating models with high variance

# The Model Confidence Set

- When the algorithm ends, the final set $\widehat{\mathcal{M}}_{1-\alpha}$ has the property

$$\lim_{P\rightarrow\infty}\Pr\left(\mathcal{M}^{\star}\subset\widehat{\mathcal{M}}_{1-\alpha}\right)\geq 1-\alpha$$

- The result follows directly since the FWE is $\leq\alpha$
- If there is only 1 “best” model, then the result can be strengthened

$$\lim_{P\rightarrow\infty}\Pr\left(\mathcal{M}^{\star}\subset\widehat{\mathcal{M}}_{1-\alpha}\right)=1$$

  - The MCS will find the “best” model asymptotically
  - The intuition behind this is that the “best” model will have:
    - Lower loss than all other models
    - The variance of the average loss differential will decline as $P\rightarrow\infty$
- When 2 or more models are equally good, there is always a $\alpha$ chance that at least 1 will be rejected
- In large samples, models which are not in $\mathcal{M}^{\star}$ will be eliminated with probability 1 since the individual test statistics are consistent

# Model Confidence Set

- The MCS takes loss functions as inputs, but ultimately works on loss differentials
- Since there is no benchmark model, all loss differentials are considered

$$\delta_{ij,t} = L\left(y_{t+h}, \hat{y}_{t+h,i|t}\right) - L\left(y_{t+h}, \hat{y}_{t+h,j|t}\right)$$

- There are many pairs, and so the actual test examines whether the average loss for model $j$ is different from that of all models

$$\bar{\delta}_i = \frac{1}{m-1} \sum_{i=1, i \neq j}^{m} \bar{\delta}_{ij}$$

- If $\bar{\delta}_i$ is sufficiently positive, then model $i$ is worse then the other models in the set

# Null and Alternative

- The MCS can be based on two test statistics
- Both satisfy some technical conditions on $d_{\mathcal{M}}$ and $e_{\mathcal{M}}$
- The first is based on $T = \max_{i\in\mathcal{M}}(\bar{z}_i)$ where $\bar{z}_i = \bar{\delta}_i/\hat{\sigma}_i$ and $\hat{\sigma}_i^2$ is an estimate of the (log-run) variance of $\bar{\delta}_i$
  - The elimination rule is $e_{\mathcal{M}} = \operatorname{argmax}_{i\in\mathcal{M}} z_i$
- The second is based on $T_R = \max_{i,j\in\mathcal{M}} \left|\bar{z}_{ij}\right|$ where $\bar{z}_{ij} = \bar{\delta}_{ij}/\hat{\sigma}_{ij}$ and $\hat{\sigma}_{ij}$ is an estimate of the (log-run) variance of $\bar{\delta}_{ij}$
  - The elimination rule is $e_{R,\mathcal{M}} = \operatorname{argmax}_{i\in\mathcal{M}} \sup_{j\in\mathcal{M}} \bar{z}_{ij}$
  - Eliminate the model which has the largest loss differential to some other model, relative to its standard deviation
- At each step the null is $H_0 : \mathcal{M} = \mathcal{M}^\star$ and the alternative is $H_1 : \mathcal{M} \supsetneq \mathcal{M}^\star$

# Model Confidence Set Setup

## Algorithm (Model Confidence Set Components)

1. *Construct a set of bootstrap indices which will be reused throughout the MCS construction using a bootstrap appropriate for the data*
2. *Construct the average loss for each model*

$$\bar{L}_j = P^{-1} \sum_{t=R+1}^{T} L_{j,t}$$

*where* $L_{j,t} = L\left(y_{t+h}, \hat{y}_{t+h,j|t}\right)$

3. *For each bootstrap replication, compute centered the bootstrap average loss*

$$\eta^{\star}_{b,j} = P^{-1} \sum_{t=R+1}^{T} L^{*}_{b,j,t} - \bar{L}_j$$

# Model Confidence Set

## Algorithm (Model Confidence Set)

1. *Being with* $\mathcal{M} = \mathcal{M}_0$ *containing all models where* $m$ *is the number of models in* $\mathcal{M}$
2. *Calculate* $\bar{L} = m^{-1} \sum_{j=1}^{m} \bar{L}_j$, $\eta_b^\star = m^{-1} \sum_{j=1}^{m} \eta_{b,j}^\star$, *and* $\hat{\sigma}_j^2 = B^{-1} \sum_{b=1}^{B} \left(\eta_{b,j}^\star - \bar{\eta}_j^\star\right)^2$ *where* $\bar{\eta}_j^\star$ *is the average of* $\eta_{b,j}^*$ *for model* $j$
3. *Define* $T = \max_{j \in \mathcal{M}} \left(\bar{z}_j\right)$ *where* $\bar{z}_j = \bar{L}_j / \hat{\sigma}_j$
4. *For each bootstrap sample, compute* $T_b^\star = \max_{j \in \mathcal{M}} \left(\left(\bar{L}_{b,j}^\star - \bar{L}_b^\star\right) / \hat{\sigma}_j\right) = \max_{j \in \mathcal{M}} \left(\left(\eta_{b,j}^\star - \eta_b^\star\right) / \hat{\sigma}_j\right)$
5. *Compute the p-value of* $\mathcal{M}$ *as* $\hat{p} = B^{-1} \sum_{b=1}^{B} I\left[T_b^\star > T\right]$
6. *If* $\hat{p} > \alpha$ *stop*
7. *If* $\hat{p} < \alpha$, *set* $e_{\mathcal{M}} = \operatorname{argmax}_{j \in \mathcal{M}} \left(\bar{z}_j\right)$ *and eliminate the model with the largest test statistic from* $\mathcal{M}$
8. *Return to step 2, using the reduced model set*

# Comments

- It is important that the variance estimates are re-computed in each step of algorithm
- This allows the standard errors to decline if poor models are excluded since the cross-sectional variance of $\bar{L}_j$ should be smaller when a bad model is dropped
- In practice the MCS should be implemented by computing in order
  1. A set of bootstrap indices
  2. The $P$ by $m$ set of bootstrapped losses $L^*_{b,j,t}$
  3. The 1 by $m$ vector containing $\eta^\star_{b,j}$
- By iterating over these $B$ times only the $B$ by $m$ matrix containing $\eta^\star_{b,j}$ has to be retained
  - Plus the 1 by $m$ vector containing $\bar{L}_j$

# Model Confidence P-value

- The MCS can also provide p-values for each model
- If model $i$ is eliminated, then the p-value of model $i$ is the maximum of the $\hat{p}$ found when model $i$ is eliminated and *all previous p-values*
- Suppose $\alpha = .05$, and the first three rounds eliminated models with $\hat{p}$ of .01,.04,.02, respectively
- The three p-values would then be:
    - $.01$(nothing to compare against)
    - $.04 = \max(.01, .04)$
    - $.04 = \max(.02, .04)$
- The output of the MCS algorithm is $\widehat{\mathcal{M}}_{1-\alpha}$ which contains the true set of best models with probability weakly larger than $1 - \alpha$
- This is similar to a standard frequentist confidence interval which contains the true parameter with probability of at least $1 - \alpha$
- The MCS p-value is not a statement about the probability that a model is the best
    - For example, the model with the lowest loss always has p-value = 1

# Model Confidence P-value

Table 1: Computation of MCS $p$-values

| Elimination Rule | $p$-value for $H_{0,\mathcal{M}_k}$ | MCS $p$-value |
|---|---|---|
| $e_{\mathcal{M}_1}$ | $P_{H_{0,\mathcal{M}_1}} = 0.01$ | $\hat{p}_{e_{\mathcal{M}_1}} = 0.01$ |
| $e_{\mathcal{M}_2}$ | $P_{H_{0,\mathcal{M}_2}} = 0.04$ | $\hat{p}_{e_{\mathcal{M}_2}} = 0.04$ |
| $e_{\mathcal{M}_3}$ | $P_{H_{0,\mathcal{M}_3}} = 0.02$ | $\hat{p}_{e_{\mathcal{M}_3}} = 0.04$ |
| $e_{\mathcal{M}_4}$ | $P_{H_{0,\mathcal{M}_4}} = 0.03$ | $\hat{p}_{e_{\mathcal{M}_4}} = 0.04$ |
| $e_{\mathcal{M}_5}$ | $P_{H_{0,\mathcal{M}_5}} = 0.07$ | $\hat{p}_{e_{\mathcal{M}_5}} = 0.07$ |
| $e_{\mathcal{M}_6}$ | $P_{H_{0,\mathcal{M}_6}} = 0.04$ | $\hat{p}_{e_{\mathcal{M}_6}} = 0.07$ |
| $e_{\mathcal{M}_7}$ | $P_{H_{0,\mathcal{M}_7}} = 0.11$ | $\hat{p}_{e_{\mathcal{M}_7}} = 0.11$ |
| $e_{\mathcal{M}_8}$ | $P_{H_{0,\mathcal{M}_8}} = 0.25$ | $\hat{p}_{e_{\mathcal{M}_8}} = 0.25$ |
| $\vdots$ | $\vdots$ | $\vdots$ |
| $e_{\mathcal{M}_{(m_0)}}$ | $P_{H_{0,\mathcal{M}_{m_0}}} \equiv 1.00$ | $\hat{p}_{e_{\mathcal{M}_{m_0}}} = 1.00$ |

# Model Confidence Set using $T_R$

## Algorithm (Model Confidence Set Components)

1. *Construct a set of bootstrap indices which will be reused throughout the MCS construction using a bootstrap appropriate for the data*
2. *Construct the average loss for each model* $\bar{L}_j = P^{-1}\sum_{t=R+1}^{T} L_{j,t}$ *where* $L_{j,t} = L\left(y_{t+h}, \hat{y}_{t+h,j|t}\right)$
3. *For each bootstrap replication, compute centered the bootstrap average loss*

$$\bar{L}^{\star}_{b,j} = P^{-1}\sum_{t=R+1}^{T} L^{*}_{b,j,t} - \bar{L}_j$$

4. *Calculate*

$$\hat{\sigma}^2_{ij} = B^{-1}\sum_{b=1}^{B}\left(\left(\bar{L}^{\star}_{b,i} - \bar{L}^{\star}_{i}\right) - \left(\bar{L}^{\star}_{b,j} - \bar{L}^{\star}_{j}\right)\right)^2$$

*where* $\bar{L}^{\star}_{j}$ *is the average of* $\bar{L}^{\star}_{b,j}$ *for the model* $j$ *across all bootstraps*

# Model Confidence Set

## Algorithm (Model Confidence Set)

1. *Being with $\mathcal{M} = \mathcal{M}_0$ containing all models where m is the number of models in $\mathcal{M}$*
2. *Define $T_R = \max_{i,j \in \mathcal{M}} \left( \bar{z}_{ij} \right)$ where $\bar{z}_{ij} = \left| \bar{L}_i - \bar{L}_j \right| / \hat{\sigma}_{ij}$*
3. *For each bootstrap sample, compute $T^{\star}_{R,b} = \max_{i,j \in \mathcal{M}} \left( \left| \bar{L}^{\star}_i - \bar{L}^{\star}_j \right| / \hat{\sigma}_{ij} \right)$*
4. *Compute the p-value of $\mathcal{M}$ as*

$$\hat{p} = B^{-1} \sum_{b=1}^{B} I \left[ T^{\star}_{R,b} > T_R \right]$$

5. *If $\hat{p} > \alpha$ stop*
6. *If $\hat{p} < \alpha$, set $e_{\mathcal{M}} = \operatorname{argmax}_{i \in \mathcal{M}} \sup_{j \in \mathcal{M}} \left( \bar{z}_{ij} \right)$ and eliminate the model with the largest test statistic from $\mathcal{M}$*
7. *Return to step 2, using the reduced model set*

# Comments

- The main difference is that the variance is *not* re-estimated in each iteration
- This happens since $T_R$ is based on the maximum DMW test statistic in each iteration
    - DMW only depends on the properties of the pair
- However, the bootstrapped distribution does depend on which models are included and so this will vary across the iterations
- This version of the algorithm requires storing the $B$ by $m$ matrix of $\bar{L}_j^{\star}$

# Confidence sets for ICs

- The MCS can be used to construct confidence sets for ICs
- This type of comparison does not directly use forecasts, and so is in-sample
- This differs from traditional model selection where only the model with the best IC is chosen
- The MCS for an IC could be used as a pre-filtering mechanism prior to combining
- Implementing the MCS on an IC is slightly more complicated than the default MCS since it is necessary to jointly bootstrap the vector $\left\{y_t, \mathbf{x}_{j,t}\right\}$ where $\mathbf{x}_{j,t}$ are the regressors in model $j$
- Paper recommends using $T_R$ statistic to compare models using $IC$
- The object of interest is

$$IC_j = T \ln \hat{\sigma}_j^2 + c_j$$

- $c_j$ is the penalty term
  - AIC: $2k_j$, BIC: $k_j \ln T$
  - AIC$^\star$: $2k_j^\star$, BIC$^\star$: $k_j^\star \ln T$
- $k_j^\star$ is known as *effective degrees of freedom* (in mis-specified model $k^\star \neq k$)
- MCS paper discusses how to estimate $k^\star$

# Confidence sets for ICs

- Using $T_R$ MCS construction algorithm, the test statistic is based on

$$T_R = \max_{i,j \in \mathcal{M}} \left| \left[ T \ln \hat{\sigma}_i^2 + c_i \right] - \left[ T \ln \hat{\sigma}_j^2 + c_j \right] \right|$$

- The bootstrap critical values are computed from

$$T_{R,b}^{\star} = \max_{i,j \in \mathcal{M}} \left( \left[ T \ln \hat{\sigma}_i^{2\star} + c_i - T \ln \hat{\sigma}_i^2 \right] - \left[ T \ln \hat{\sigma}_j^{2\star} + c_j - T \ln \hat{\sigma}_j^2 \right] \right)$$

- $\hat{\sigma}_i^{2\star}$ is the variance computed using

$$\epsilon_{b,t}^{\star} = y_{b,t}^{\star} - \mathbf{x}_{b,j,t}^{\star\prime} \hat{\boldsymbol{\beta}}_{b,j}^{\star}$$

- $\hat{\boldsymbol{\beta}}_{b,j}^{\star}$ is re-estimated using the bootstrapped data $\left\{ y_{b,t}^{\star}, \mathbf{x}_{b,j,t}^{\star} \right\}$
- Errors are computed using the bootstrapped data and parameter estimates
- Aside from these changes, the remainder of the algorithm is unmodified

# False Discovery Rate and FWER

- Controlling False Discover Rate (FDR) is an alternative to controlling Family Wise Error Rate (FWER)

## Definition ($k$-Familywise Error Rate)

For a set of null and alternative hypotheses $H_{0,i}$ and $H_{1,i}$ for $i = 1, \ldots, m$, let $\mathcal{I}_0$ contain the indices of the correct null hypotheses. The $k$-Familywise Error Rate is defined as

$$\Pr\left(\text{Rejecting at least } k\ H_{0,i} \text{ for } i \in \mathcal{I}_0\right) = 1 - \Pr\left(\text{Reject no } H_{0,i} \text{ for } i \in \mathcal{I}_0\right)$$

- $k$ is typically 1, so the testing procedures control the probability of any number of false rejections
  - Type I errors
- The makes FWER tests possibly conservative
  - Depends on what the actual intent of the study is

# False Discovery Rate

## Definition

The False Discovery Rate is the percentage of false null hypothesis relative to the total number of rejections, and is defined

$$FDR = {}^{F}\!/_{R}$$

where $F$ is the number of false rejections and $R$ is the total number of rejections.

- Unlike FWER, methods that control FDR explicitly assume that some rejections are false.
- Ultimately this leads to a (potentially) procedure that might discover more actual rejections
- For standard DMW-type tests, both FWER and FDR control fundamentally reduce to choosing a critical value different from the usual $\pm 1.96$
  - Most of the time larger in magnitude
  - Can be smaller in the case of FDR when there are many false nulls

# False Discovery Rate

- FDR is naturally *adaptive*
- When the number of false nulls is small (~0), then FDR should choose a critical value similar to the FWER-based procedures
    - $R \approx F$, $F/R \approx 1$ so any $F$ is too large
    - On the other hand, when the percentage of false nulls is near 100%, can reject all nulls
        - $F \approx 0$, $F/R \approx 0$ and all nulls can be rejected
        - Critical value can be arbitrarily small since virtually no tests have small values
        - Hypothetically, could have a critical value of 0 if all nulls were actually false
- FDR controls the false rejection rate, and it is common to use rates in the range of 5-10%
    - Ultimately should depend on risk associated with trading a bad strategy against the cost of missing a good strategy
    - Adding a small percentage of near 0 excess return strategies to a large set of useful strategies shouldn't deteriorate performance substantially

# Operationalizing FDR

UNIVERSITY OF OXFORD

- Operationalizing FDR requires some estimates
- In standard trading strategy setup, $H_0 : \mu = 0$, $H_A : \mu \neq 0$ where $\mu$ is the expected return in excess of some benchmark
  - Benchmark might be risk-free rate, or could be buy-and-hold strategy
- $\pi$ is the proportion of false nulls
  - Estimated using information about the distribution of p-values “near” 1 since these should all be generated from true nulls
  - Entire procedure relies on only p-values
    - Similar to Bonferoni or Bonferoni-Holm
  - For standard 2-sided alternative

$$p_i = 2\left(1 - \Phi\left(|t_i|\right)\right)$$

where $t_i$ is (normalized) test statistic for strategy $i$.

# Computing FDR

- Key idea is to find $\gamma$, which is some number in $[0, 1]$ such that

$$\alpha = \widehat{FDR} \equiv \frac{\hat{\pi} l \gamma}{\sum_{i=1}^{l} I\left[p_i < \gamma\right]}$$

- where
  - $\alpha$ is the target FDR rate
  - $\hat{\pi}$ and an estimate of the percentage of nulls that are true (no abnormal performance)
  - $l$ is the number of rules
  - $\gamma$ is the parameter that is used to find the p-value cutoff
  - $\sum_{i=1}^{l} I\left[p_i < \gamma\right]$ is the number of rejections using $\gamma$
- The numerator is simply an estimate of the number of false rejections, which is
  Probability of Null True $\times$ Number of Hypotheses = Number of True Hypotheses
  Number of False Hypotheses $\times$ Cutoff = Number of False that are Rejected using $\gamma$
- Exploits the fact that under the null p-values have a uniform distribution, so that if there are $M$ false nulls, then, using a threshold of $\gamma$ will reject $\gamma M$

# Positive and Negative FDR

- Can further decompose FDR into upper (better) and lower (worse) measures

$$\widehat{FDR}^{+} \equiv \frac{{}^{1}\!/\!{}_{2}\hat{\pi} l \gamma_U}{\sum_{i=1}^{l} I\left[p_i < \gamma_U, t_i > 0\right]}, \quad \widehat{FDR}^{+} \equiv \frac{{}^{1}\!/\!{}_{2}\hat{\pi} l \gamma_L}{\sum_{i=1}^{l} I\left[p_i < \gamma_L, t_i < 0\right]}$$

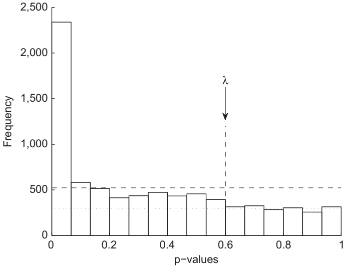
- This version assumes a symmetric 2-sided test statistic, so that on average 50% of the false rejections are in each tail
- Allows for tail-specific choice of $\gamma$ which would naturally vary if the number of correct rejections was different
  - Suppose for example that many rules were bad, then $\gamma_L$ would be relatively large

# Estimation of $\pi$

- $\pi$ is estimated as

$$\hat{\pi} = \frac{\sum_{i=1}^{l} I\left[p_k > \lambda\right]}{l\left(1-\lambda\right)}$$

- $\lambda$is a tuning parameter
  - Simple to choose using visual inspection
  - Recall that true nulls lead to a flat p-value histogram
  - Find point where histogram looks non-flat, use cutoff for $\lambda$
- Histogram from BS



# Estimating $\pi$

- $\hat{\pi}$ allows percentage of correct rejections to be computed as $\hat{\pi}^A = 1 - \hat{\pi}$
- In the decomposed FDR the number of good (bad) rules can be computed as

$$\alpha \times \sum_{i=1}^{l} I\left[p_i < \gamma_U, t_i > 0\right]$$

  - Note that $\gamma_U$ is fixed here

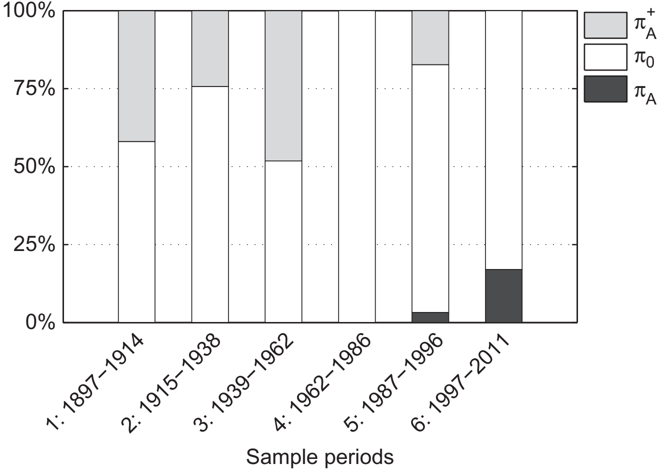
# Bajgrowicz & Scaillet (*JFE*, 2012)

- Apply FDR to technical trading rules of STW
- Use DJIA
  - 1897-2011
- Find similar results, although importantly consider transaction costs for break even
  - Strategies that trade more can have higher means while not violating EMH

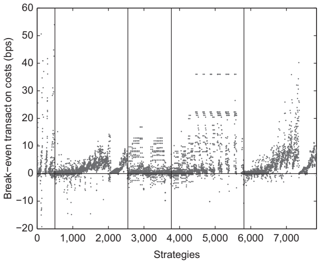
# Background on Competitor Methods

| Sample period | RW portfolio | | Best rule | | DJIA |
|---|---|---|---|---|---|
| | Sharpe ratio | Portfolio size | Sharpe ratio | BRC *p*-value | Sharpe ratio |
| 1: 1897–1914 | 1.24 | 45 | 1.18 | 0.00 | −0.12 |
| 2: 1915–1938 | – | 0 | 0.73 | 0.11 | 0.06 |
| 3: 1939–1962 | 1.49 | 62 | 2.34 | 0.00 | 0.41 |
| 4: 1962–1986 | 1.52 | 15 | 1.45 | 0.00 | −0.16 |
| 5: 1987–1996 | – | 0 | 0.84 | 0.93 | 0.66 |
| 6: 1997–2011 | – | 0 | 0.48 | 1.00 | 0.12 |
| 1897–1996 | 0.70 | 88 | 0.82 | 0.00 | 0.12 |

# Good and Bad Rules



# Transaction Costs Required for 0-profit (-1962)

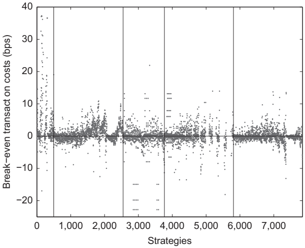


- Transaction costs are important when assessing rules
- Rather than apply arbitrary TC, look for break even
- Transaction costs are a function of mean and number of transactions

$$0 = \mu_i - TC \times \#\{trades\}$$

- $\mu_i$ is the full-sample mean, not the annualized

# Transaction Costs Required for 0-profit (1962-)



- Transaction for break even are lower
- Actual transaction costs are lower
- Unclear whether this is driven by more trading signals or worse mean

# Persistence of Rules

| Sample period | FDR portfolio | | | RW portfolio | | | 50 best rules | | Best rule | |
|---|---|---|---|---|---|---|---|---|---|---|
| | IS | OOS | Median size | IS | OOS | Median size | IS | OOS | IS | OOS |
| 1: 1897–1914 | 3.41 | 0.47 | 14 | 1.31 | 0.51 | 0 | 5.79 | 0.50 | 6.34 | 0.03 |
| 2: 1915–1938 | 4.62 | 0.01 | 13 | 0.90 | 0.17 | 0 | 5.39 | −0.03 | 5.98 | 0.09 |
| 3: 1939–1962 | 4.77 | 0.55 | 15 | 1.85 | 0.09 | 0 | 5.78 | 0.43 | 6.70 | 0.12 |
| 4: 1962–1986 | 5.34 | −0.31 | 13 | 1.36 | 0.14 | 0 | 6.17 | −0.18 | 6.95 | −0.59 |
| 5: 1987–1996 | 4.52 | −0.34 | 12 | – | – | – | 5.44 | −0.37 | 6.07 | 0.08 |
| 6: 1997–2011 | 4.55 | −0.74 | 12 | 0.78 | 0.07 | 0 | 5.22 | −0.51 | 5.97 | −0.27 |

- Sharpe-Ratios
- Persistence is low
- Conservative Romano-Wolf appears to have more persistence
- Combination appears to be not help