DM812
METAHEURISTICS

Lecture 4

Empirical Methods for Configuring and Tuning

Marco Chiarandini

Department of Mathematics and Computer Science
University of Southern Denmark, Odense, Denmark
<marco@imada.sdu.dk>

Outline

Outline

Statistical Tests

Univariate Analysis: On a Class of Instances

Global tests	Unreplicated	Replicated
Parametric	F-test	F-test
Non-Parametric Rank based	Friedman Test	Friedman Test
Non-Parametric Permutation based	Simple Permutations	Synchronized Permutations

Statistical Tests
Univariate Analysis: On a Class of Instances

Pairwise tests	Unreplicated	Replicated
Parametric	t-test Tukey HSD	t-test Tukey HSD
Non-Parametric Rank based	Friedman Test or *Wilcoxon Signed Rank Test*	Friedman Test
Non-Parametric Permutation based	Simple Permutations	Synchronized Permutations

- Matched pairs versions: when, when not
- t-test Welch variant: no assumption of equal variances

ANOVA

Single Factor Analysis of Variance

$$X_{it} = \mu_i + \epsilon_{it}$$

$$X_{it} = \mu + \alpha_i + \epsilon_{it}$$

$$\sum_{i=1}^{k}\sum_{t=1}^{r}(X_{it} - \bar{X}_{..})^2 = \sum_{i=1}^{k}\sum_{t=1}^{r}(X_{it} - \bar{X}_{i.})^2 + \sum_{i=1}^{k} r(\bar{X}_{i.} - \bar{X}_{..})^2$$

(decomposition in *within-group* and *between-group* sum of squares)

$$MST = \frac{\sum_{i=1}^{k} r\left(\bar{X}_{i.} - \bar{X}_{..}\right)^2}{k-1} \qquad MSE = \frac{\sum_{i=1}^{k}\sum_{t=1}^{r}\left(X_{it} - \bar{X}_{i.}\right)^2}{N-k}$$

(MST *mean square per treatment* and MSE *mean square per error*)

$$F = \frac{MST}{MSE}$$

$F \sim F_{k-1,r-k}$, ie, the F-ratio approximates a the Fisher distribution F_{df_1,df_2} with df_1 and df_2 degrees of freedom

ANOVA
Two Factors Analysis of Variance

Model:

$$X_{hi} = \mu + \alpha_i + \theta_h + \epsilon_{hi}$$

Mean square:

$$MST = \frac{b\sum_{i=1}^{k}(\bar{X}_{.i} - \bar{X}_{..})^2}{k-1}; \quad MSE = \frac{\sum_{h=1}^{b}\sum_{i=1}^{k}\left(X_{hi} - \bar{X}_{h.} - \bar{X}_{.i} + \bar{X}_{..}\right)^2}{bk - b - k + 1}$$

Statistical Tests

- ANOVA through F-ratio and Fisher test
- Friedman test

ANOVA

Two Factors Repeated Measures Analysis of Variance

Model:

$$X_{hit} = \mu + \alpha_i + \theta_h + \epsilon_{hit}$$

or alternatively:

$$X_{hit} = \mu + \alpha_i + \theta_h + \alpha\theta_{hi} + \epsilon_{hit}$$

Statistical Tests

- ANOVA through F-ratio and Fisher test
- Friedman test

If the interactions between factors are of interest interaction plots are useful to visualize them

An Example

SLS algorithms for Graph Coloring:
Results collected on a set of benchmark instances

Instance	HEA		TS_{N1}		ILS		MinConf		XRLF	
Instance	Succ.	k	Succ.	k	Succ.	k	Succ.	k	Succ.	k
flat300_20_0	10	20	10	20	10	20	10	20	6	20
flat300_26_0	10	26	10	26	10	26	10	26	1	33
flat300_28_0	6	31	4	31	2	31	1	31	1	34
flat1000_50_0	4	50	2	85	6	88	4	87	1	84
flat1000_60_0	4	87	3	88	1	89	4	89	6	87
flat1000_76_0	1	88	1	88	1	89	8	90	6	87
	GLS		SA_{N2}		Novelty		TS_{N3}			
Instance	Succ.	k	Succ.	k	Succ.	k	Succ.	k		
flat300_20_0	10	20	10	20	1	22	1	33		
flat300_26_0	10	33	1	32	4	29	6	35		
flat300_28_0	8	33	8	33	10	35	4	35		
flat1000_50_0	10	50	1	86	6	54	1	95		
flat1000_60_0	4	90	1	88	4	64	1	96		
flat1000_76_0	8	92	4	89	8	98	1	96		

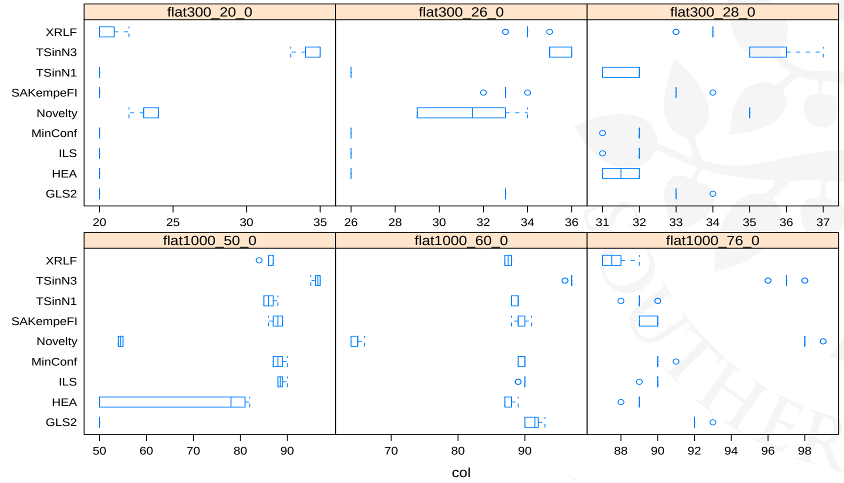

```
> load("gcp-all-classes.dataR")
> G <- F[F$class == "Flat", ]
> str(G)
'data.frame': 540 obs. of  8 variables:
 $ alg  : Factor w/ 9 levels "GLS2","HEA","ILS",..: 7 2 3 4 1 5 6 8 9 7 ...
 $ run  : Factor w/ 10 levels "1","10","2","3",..: 1 1 1 1 1 1 1 1 3 ...
 $ inst : Factor w/ 80 levels "1-FullIns_3",..: 42 42 42 42 42 42 42 42 42 42 .
 $ col  : int  85 50 88 89 50 55 86 96 84 85 ...
 $ rank : num  33 7 62 73.5 7 21 39.5 83.5 31 33 ...
 $ err2 : num  3.47 1.63 3.63 3.68 1.63 ...
 $ err3 : num  0.626 0.294 0.655 0.664 0.294 ...
 $ class: Factor w/ 12 levels "FullIns","Insertions",..: 7 7 7 7 7 7 7 7 7 7 ..
```



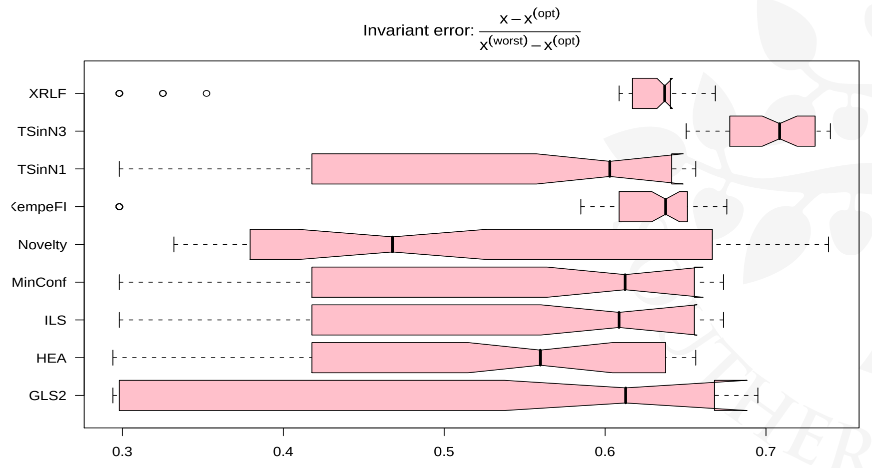
```
> with(G, table(alg, factor(inst)))
alg         flat1000_50_0 flat1000_60_0 flat1000_76_0 flat300_20_0 flat300_26_0
  GLS2                 10            10            10           10           10
  HEA                  10            10            10           10           10
  ILS                  10            10            10           10           10
  MinConf              10            10            10           10           10
  Novelty              10            10            10           10           10
  SAKempeFI            10            10            10           10           10
  TSinN1               10            10            10           10           10
  TSinN3               10            10            10           10           10
  XRLF                 10            10            10           10           10
alg         flat300_28_0
  GLS2                10
  HEA                 10
  ILS                 10
  MinConf             10
  Novelty             10
  SAKempeFI           10
  TSinN1              10
  TSinN3              10
  XRLF                10
```



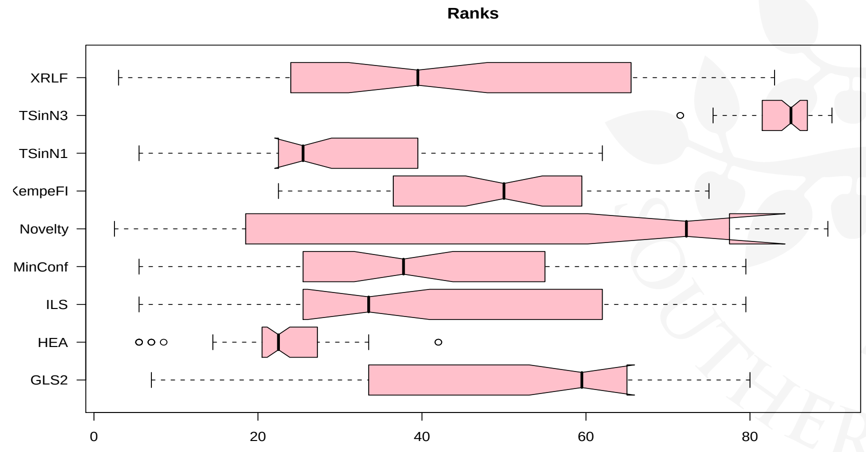
```
> print(bwplot(alg ~ col | inst, data = G, scales = list(x = list(relat
+     pch = "|"))
```



```
> boxplot(err3 ~ alg, data = G, horizontal = TRUE, main = expression(pa
+     frac(x - x^(opt), x^(worst) - x^(opt))), notch = TRUE, col = "pi
```




```
> boxplot(rank ~ alg, data = G, horizontal = TRUE, main = "Ranks",
+     notch = TRUE, col = "pink")
```



Parametric Analysis

F-test

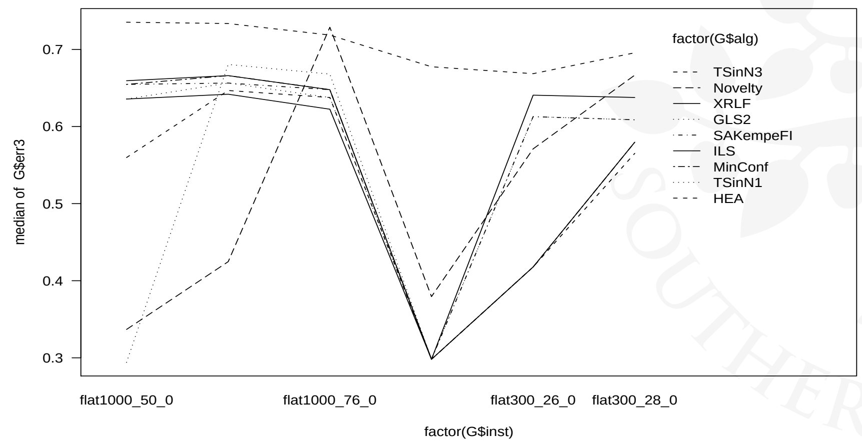

```
> l <- lm(err3 ~ as.factor(alg) * as.factor(inst), data = G)
anova(l)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
as.factor(alg)	8	1.71	0.21	464.96	0.0000
as.factor(inst)	5	5.72	1.14	2484.44	0.0000
as.factor(alg):as.factor(inst)	40	3.38	0.08	183.52	0.0000
Residuals	486	0.22	0.00		

Parametric Analysis

F-test, Interaction plots

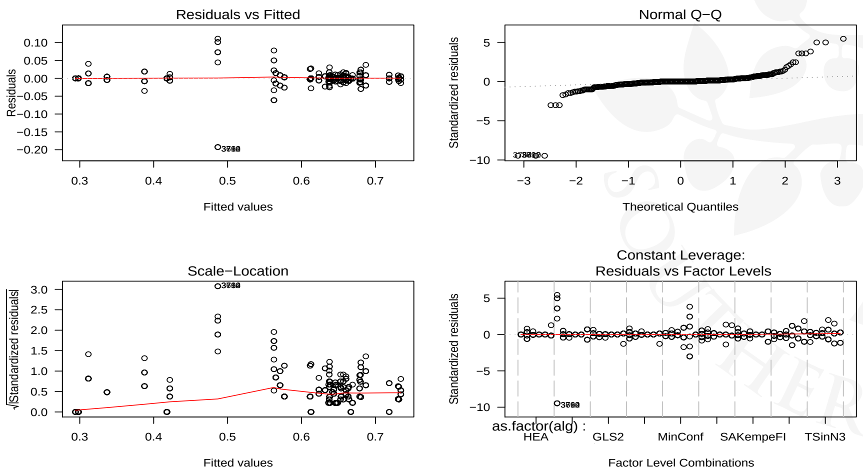

```
> interaction.plot(factor(G$inst), factor(G$alg), G$err3, median)
```



Parametric Analysis

F-test, Check Assumptions

```
> l <- lm(err3 ~ as.factor(alg) * as.factor(inst), data = G)
> par(mfrow = c(2, 2))
> plot(l)
```



Parametric Analysis

Post hoc: t test

```
> G1 <- G[G$run == 1, ]
> pairwise.t.test(G1$err3, G1$alg, paired = TRUE, var.equal = FALSE,
+     pool.sd = FALSE, p.adjust.method = "holm")
```

Pairwise comparisons using paired t tests
data: G1$err3 and G1$alg

	GLS2	HEA	ILS	MinConf	Novelty	SAKempeFI	TSinN1	TSinN3
HEA	1.00	-	-	-	-	-	-	-
ILS	1.00	1.00	-	-	-	-	-	-
MinConf	1.00	1.00	1.00	-	-	-	-	-
Novelty	1.00	1.00	1.00	1.00	-	-	-	-
SAKempeFI	1.00	1.00	1.00	1.00	1.00	-	-	-
TSinN1	1.00	1.00	1.00	1.00	1.00	1.00	-	-
TSinN3	1.00	0.63	1.00	1.00	1.00	1.00	0.82	-
XRLF	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

P value adjustment method: holm

Parametric Analysis

Post hoc: t test

```
> pairwise.t.test(G$err3, G$alg, paired = TRUE, var.equal = FALSE,
+     pool.sd = FALSE, p.adjust.method = "holm")
```

Pairwise comparisons using paired t tests
data: G$err3 and G$alg

	GLS2	HEA	ILS	MinConf	Novelty	SAKempeFI	TSinN1	TSinN3
HEA	1.00000	-	-	-	-	-	-	-
ILS	1.00000	0.02390	-	-	-	-	-	-
MinConf	1.00000	0.02390	1.00000	-	-	-	-	-
Novelty	1.00000	1.00000	1.00000	1.00000	-	-	-	-
SAKempeFI	0.06504	6.5e-06	0.00809	0.00702	0.06504	-	-	-
TSinN1	1.00000	0.07768	0.00011	0.00017	1.00000	0.00026	-	-
TSinN3	7.4e-10	< 2e-16	2.8e-13	2.6e-13	1.7e-11	4.8e-10	8.2e-15	-
XRLF	0.04779	2.2e-05	0.03549	0.03274	0.04285	1.00000	0.00295	2.1e-10

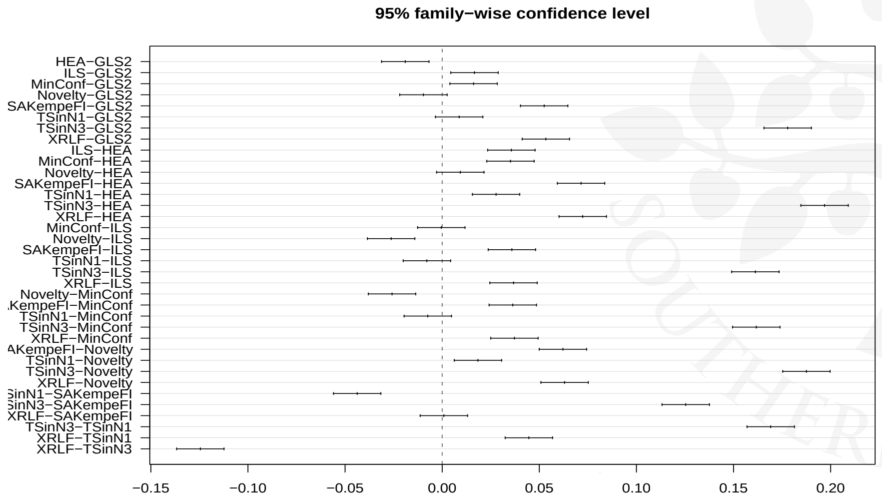
P value adjustment method: holm

Alternatively, `binom.test`

Parametric Analysis

Post hoc: Tukey HSD test

```
> par(las = 1, mar = c(3, 8, 3, 1))
> plot(TukeyHSD(aov(err3 ~ alg * inst, data = G), which = "alg"),
+     las = 1, mar = c(3, 7, 3, 1))
```



Non-Parametric Analysis

Friedman test – unreplicated

```
friedman.test              package:stats              R Documentation

Friedman Rank Sum Test

Description:

     Performs a Friedman rank sum test with unreplicated blocked data.

> G1 <- G[G$run == 1, ]
> G1$alg <- factor(G1$alg)
> G1$inst <- factor(G1$inst)
> friedman.test(formula = col ~ alg | inst, data = G1)

Friedman rank sum test
data:  col and alg and inst
Friedman chi-squared = 21.5811, df = 8, p-value = 0.005754
```


Non-Parametric Analysis

Post hoc: Wilcoxon test

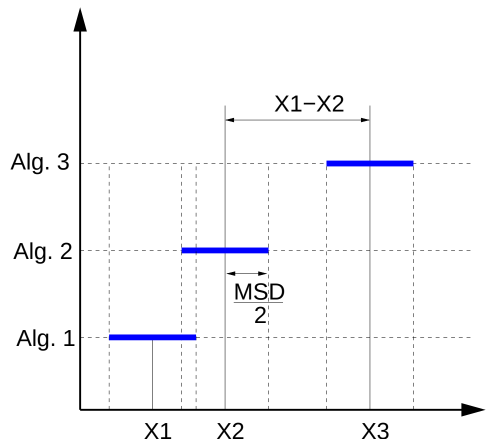
```
> pairwise.wilcox.test(G1$col, as.factor(G1$alg), paired = TRUE,
+     p.adj = "holm")

Pairwise comparisons using Wilcoxon signed rank test
data:  G1$col and as.factor(G1$alg)

          GLS2 HEA ILS MinConf Novelty SAKempeFI TSinN1 TSinN3
HEA       1    -   -   -       -       -         -      -
ILS       1    1   -   -       -       -         -      -
MinConf   1    1   1   -       -       -         -      -
Novelty   1    1   1   1       -       -         -      -
SAKempeFI 1    1   1   1       1       -         -      -
TSinN1    1    1   1   1       1       1         -      -
TSinN3    1    1   1   1       1       1         1      -
XRLF      1    1   1   1       1       1         1      1

P value adjustment method: holm
```


Non-Parametric Analysis

Post hoc: Wilcoxon test

```
> pairwise.wilcox.test(G$col, as.factor(G$alg), paired = TRUE,
+     p.adj = "holm")

Pairwise comparisons using Wilcoxon signed rank test
data:  G$col and as.factor(G$alg)

          GLS2    HEA     ILS     MinConf Novelty SAKempeFI TSinN1  TSinN3
HEA       1.00000 -       -       -       -       -         -       -
ILS       1.00000 1.00000 -       -       -       -         -       -
MinConf   1.00000 1.00000 1.00000 -       -       -         -       -
Novelty   1.00000 1.00000 1.00000 1.00000 -       -         -       -
SAKempeFI 1.00000 0.48628 1.00000 1.00000 1.00000 -         -       -
TSinN1    1.00000 1.00000 1.00000 1.00000 1.00000 1.00000   -       -
TSinN3    0.00012 7.4e-05 7.4e-05 7.4e-05 0.40106 0.00012   7.4e-05 -
XRLF      1.00000 1.00000 1.00000 1.00000 1.00000 1.00000   1.00000 0.00050

P value adjustment method: holm
```


An Example

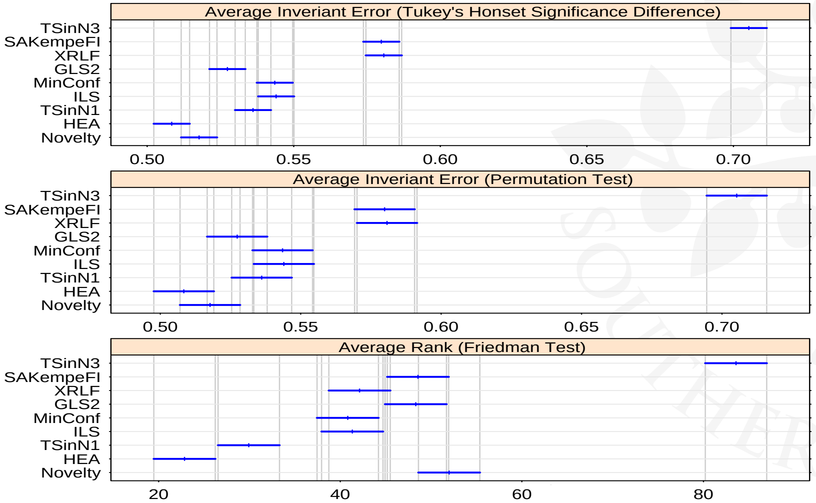


Minimal Significant Difference (MSD)

interval that satisfies simultaneously each comparison

Differences are statistically significant if the confidence intervals do not overlap

An Example



Outline

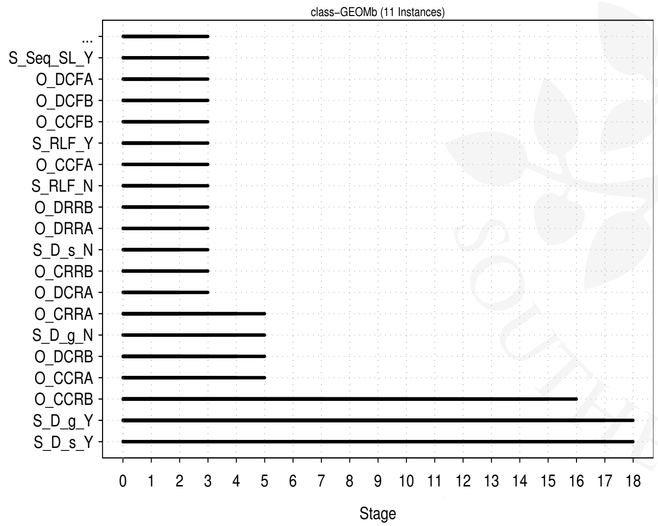
1. Inferential Statistics
   Statistical Tests for our Scenarios
   Example
2. Sequential Testing (Racing)
3. Results from DM811

Unreplicated Designs

Procedure Race [Birattari 2002]:
repeat
  Randomly select an unseen instance and run all candidates on it
  Perform *all-pairwise comparison* statistical tests
  Drop all candidates that are significantly inferior to the best algorithm
until only one candidate left or no more unseen instances ;

- F-Race use Friedman test
- Note: If resources available for N runs then the optimal design is one run on N instances

Sequential Testing



Alternatives

- ANOVA
- Regression trees [Bartz-Beielstein and Markon, 2004]
- Racing algorithms [Birattari et al., 2002]
- Search approaches
  [Minton 1993, 1996, Cavazos & O'Boyle 2005, Adenso-Diaz & Laguna 2006, Audet & Orban 2006, Hutter et al., 2007]
- Response surface models, DACE
  [Bartz-Beielstein, 2006; Ridge and Kudenko, 2007a,b]

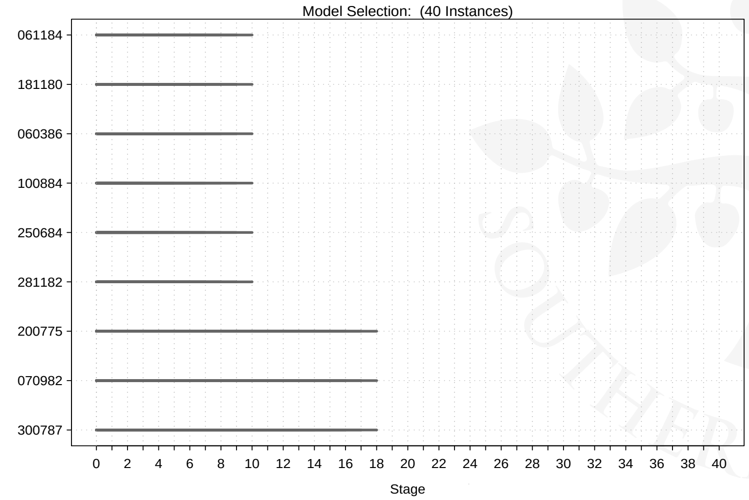
Outline

Over all instances

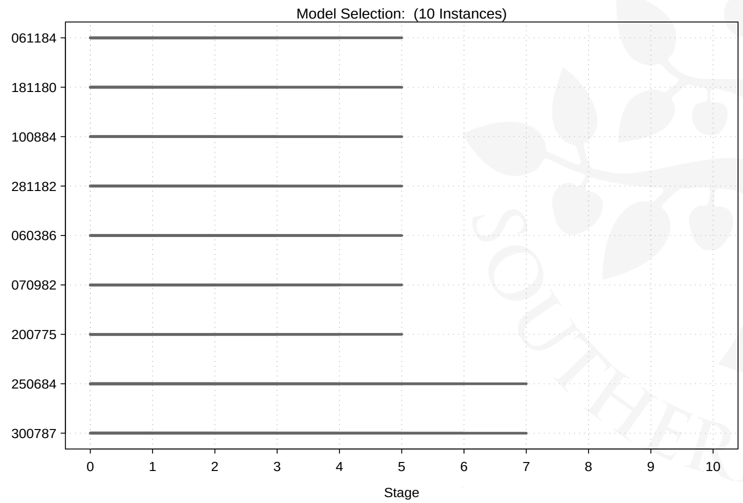
```
> O <- race("wrapper-race.R", maxExp=5000,stat.test=c("friedman"),
+ conf.level=0.95, first.test=10, interactive=TRUE,
+ log.file="race.log", no.slaves=0)
Racing methods for the selection of the best
Copyright (C) 2003 Mauro Birattari
This software comes with ABSOLUTELY NO WARRANTY

Race name.................................Model Selection:
Number of candidates.......................................9
Number of available tasks.................................40
Max number of experiments...............................5000
Statistical test................................Friedman test
Tasks seen before discarding..............................10
Initialization function...................................ok
Parallel Virtual Machine..................................no

+-+-----------+-----------+-----------+-----------+-----------+
| |       Task|      Alive|       Best|  Mean best| Exp so far|
+-+-----------+-----------+-----------+-----------+-----------+
|x|          1|          9|          4|       -924|          9|
|x|          2|          9|          1|     -826.5|         18|
...
|x|          8|          9|          1|     -874.4|         72|
|x|          9|          9|          1|     -879.9|         81|
|-|         10|          3|          1|     -865.2|         90|
|=|         11|          3|          1|     -866.5|         93|
...
|=|         17|          3|          1|     -856.1|        111|
|-|         18|          1|          1|     -849.2|        114|
+-+-----------+-----------+-----------+-----------+-----------+

Selected candidate:          1 mean value:     -849.2

Description of the selected candidate:
   label      path                     command
1 300787 300787/src Driver -tt 30 -ch 2 -ls 3
```

Inferential Statistics
Sequential Testing
Results from DM811

Over all instances

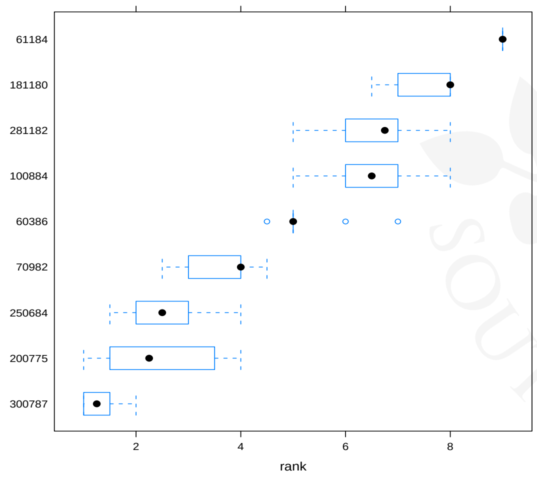


u-1000-10-1000



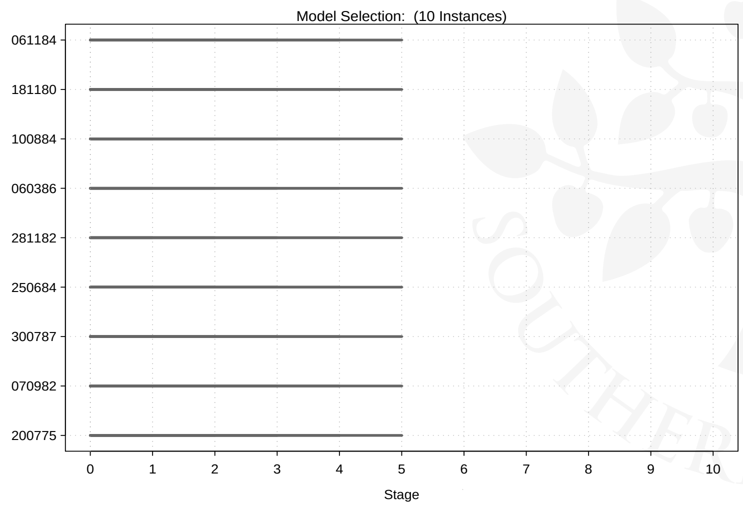
Inferential Statistics
Sequential Testing
Results from DM811

u-1000-10-1000



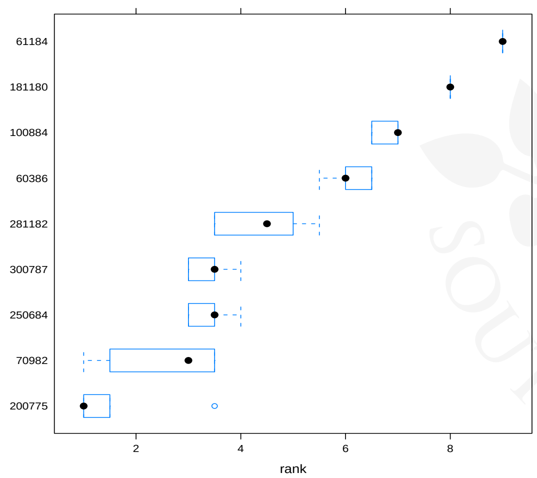
Inferential Statistics
Sequential Testing
Results from DM811

u-1000-50-1000

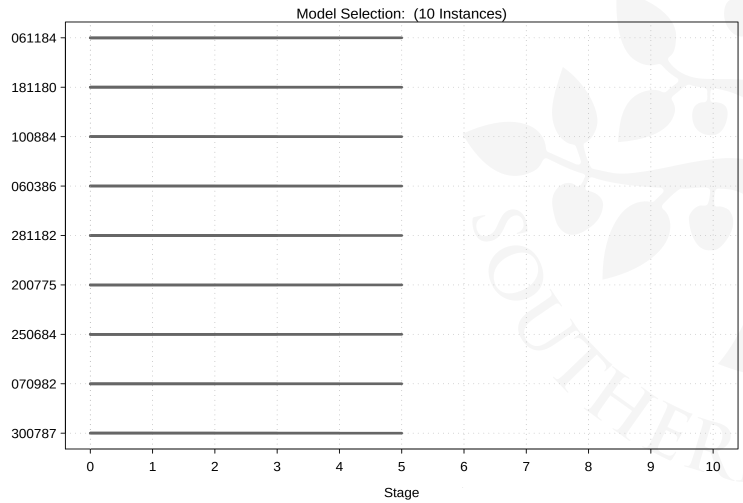


Inferential Statistics
Sequential Testing
Results from DM811

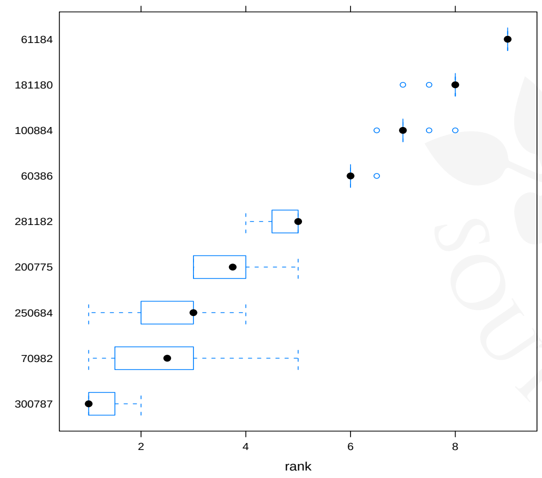
u-1000-50-1000



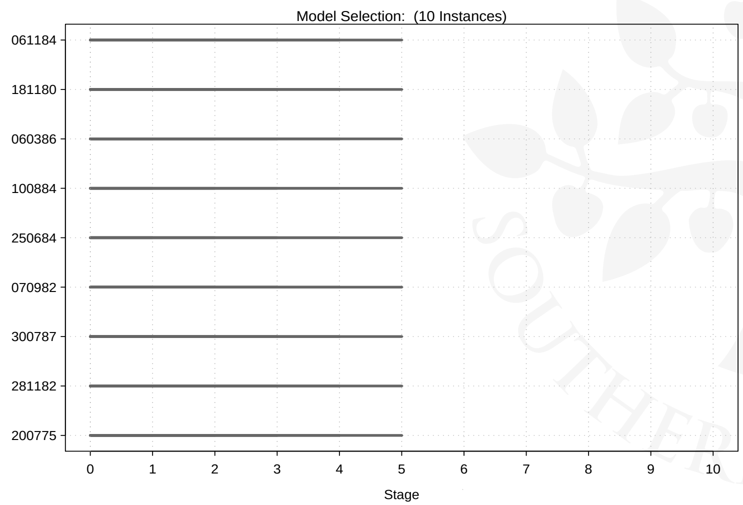
u-1000-10-10000



u-1000-10-10000



u-1000-50-10000



u-1000-50-10000