UiO : Department of Informatics
University of Oslo

Introducing Statistical Design of Experiments to SPARQL Endpoint Evaluation

Kjetil Kjernsmo and John Tyssedal

24.10.2013

Objective

- To introduce a path to critical practice of evaluations, that makes use of contemporary statistical techniques, to establish a practice that can be used to refute assertions on performance.
- The focus is didactical, and the experiment we present is a toy example.
- To apply a well established method in statistics, rarely used in Computer Science, to SPARQL endpoint evaluation.

Problems with benchmarking

- Large number of parameters to test for complex scenarios.
- No structured approach to investigate flaws.
- No meaningful summary of the overall performance.
- Standardized benchmarks cannot test assertions outside of the standard.
- Attempting to neutralize the effect of certain optimization techniques oversimplifies the test.

Design of Experiments

- Pioneered by Fischer in the 1920s.
- Well-established in many fields of engineering, agriculture, medicine.
- Well-suited to manage very complex experiments.
- Requires parameterization of experiments.

Response variable

A *response variable* is measured under various combinations of parameters.

Examples

- Throughput
- Query execution time
- Response time

Such parameters are called *factor*s.

Examples

- A concrete implementation
- Hardware platform
- Data heterogeneity
- Number of triples in the store
- Absence or presence of certain language features
- Order of experiments
- Tuning parameters
- Concurrency

Levels

For each factor, a range of possible values are fixed. These values are called *levels*.

Examples

- For implementation, 4store or Virtuoso.
- For number of triples, the levels could be 1 or 2 MTriples.
- For language features, the levels could be SELECT and CONSTRUCT

Levels may be continuous, discrete, different instances of a class, etc.

Design Matrix

	Implement	TripleC	Union
1	2	1	2
2	2	2	2
3	2	1	1
4	1	1	1
5	2	2	1
6	1	2	1
7	1	2	2
8	1	1	2

Effects describe the influence of the levels on the response.

- Similarities to regression.
- *Main effects* consider the factors on their own.
- *Interaction effects* consider factors given levels of other factors.
- In practice done with linear regression routines.

Factorial experiments

- Full factorial experiments:
  - 2 levels give 2^n combinations (*runs*), where n is the number of factors.
  - 3 levels give 3^n runs, etc.
- Fractional factorial experiments:
  - When experimental economy is important,
  - DoE theory offers extensive facilities for running a fraction of the runs (2^{n-1} etc).
  - The price is of smaller experiments is explanatory power.

Constraints on scope of experiment

Above all: Easy to understand for newcomers to DoE!

- Just 2 levels.
- Handful of illustrative factors.
- Analysis that illustrates key concepts.
- Straightforward to program and reproduce.
- Artificial performance differences.

Choices to meet these constraints

- Response variable: Time from DNS lookup finishes to endpoint has delivered a full response, measured by `curl`.
- We don't compare different implementions, we compare the performance before and after some change.
- Query engine: `4store` where we insert sleep statements in the join function on one level and language matching function on the other.

Factors and levels

“Implement”	The implementation under evaluation.
“TripleC”	1 or 2 MTriples from the DBPedia SPARQL Benchmark.
“Machine”	Software and hardware platform, one smaller machine with fast SSD and one larger with slower disks in RAID1.

The Basic Graph pattern factor

"BGPComp" Two different basic graph patterns with varying complexity.

Level 1

```
?s  rdfs:label ?l1 ;
    ?p1 ?o1 .
?o1 dbo:populationTotal ?l2 .
```

Level 2

```
?s  rdfs:label ?l1 ;
    ?p1 ?o1 .
?o1 dbo:populationTotal ?l2 .
?s foaf:page ?o2 ;
   dbpprop:subdivisionName ?o3 .
?o3 skos:subject ?o4 ;
   dbpprop:seat ?o5 ;
   a ?c1 .
```

Absence or presence factors

"Union" Absence or presence of a UNION pattern.

```
{ ?o1 dbpprop:longd ?long ;
      dbpprop:latd  ?lat .
} UNION {
  ?o1 geo:long ?long ;
      geo:lat ?lat . }
```

"Lang" Absence or presence of a langMatches filter.

```
FILTER langMatches( lang(?l1), "fr" )
```

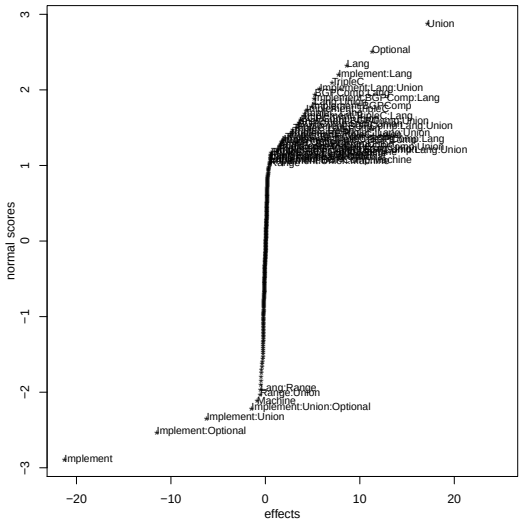
"Range" Absence or presence of a filter with a larger-than operator.

```
FILTER (?l2 > 800000)
```

"Optional" Absence or presence of an OPTIONAL pattern.

```
OPTIONAL { ?o1 foaf:homepage ?o6 . }
```

Full normal plot



Factors	Effect
Implement	−21.27
Implement:Optional	−11.49
Implement:Union	−6.21
Implement:TripleC:Lang1	4.12
TripleC:Lang	4.20
Implement:TripleC	4.32
Implement:BGPComp	4.89
Lang:Union	5.16
Implement:BGPComp:Lang	5.16
BGPComp:Lang	5.25
Implement:Lang:Union	5.75
TripleC	7.06
Implement:Lang	7.73
Lang	8.62
Optional	11.30
Union	17.18

Preparing hypothesis tests

Hypothesis formulation:

- H_0: The new implementation is no better than the old.
- H_1: The new implementation is better than the old.

Classifying factors:

- “Range” and “Machine” are called *inactive factors*.
- “BGPComp”, “Lang”, “Optional”, “Union” and “TripleC” are called *environmental factors*.
- “Implement” is called a *control factor*.

- Average over all the 32 level-combinations of the environmental factors.

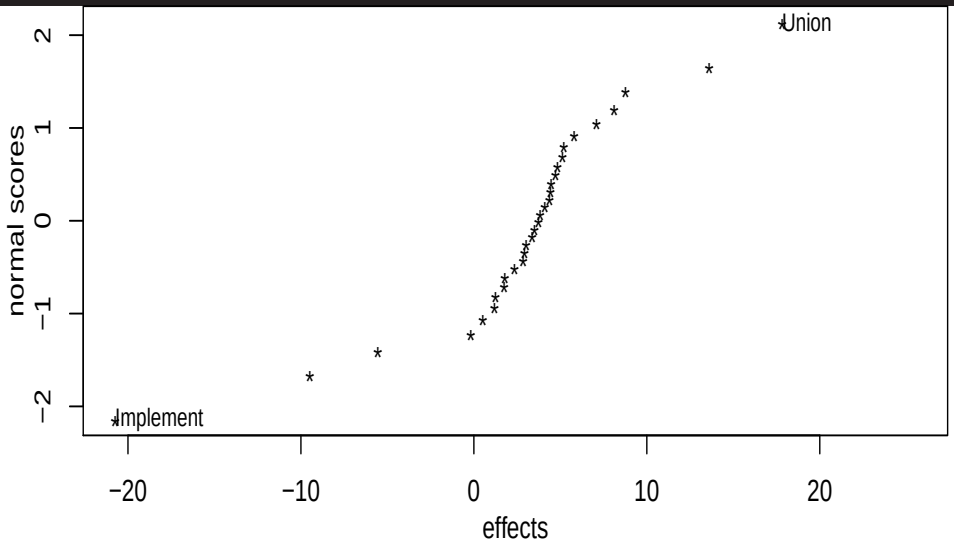
	Implement	Machine	Range	Response
1	1	1	1	33.47
2	2	1	1	11.82
3	1	2	1	32.16
4	2	2	1	11.56
5	1	1	2	33.97
6	2	1	2	12.46
7	1	2	2	32.90
8	2	2	2	11.60

- May create a one-sided two-sample t-test with 4 values for each of the levels of “Implement”.
- Lends support to H_1 with a high probability, $p = 1.16 \cdot 10^{-7}$.

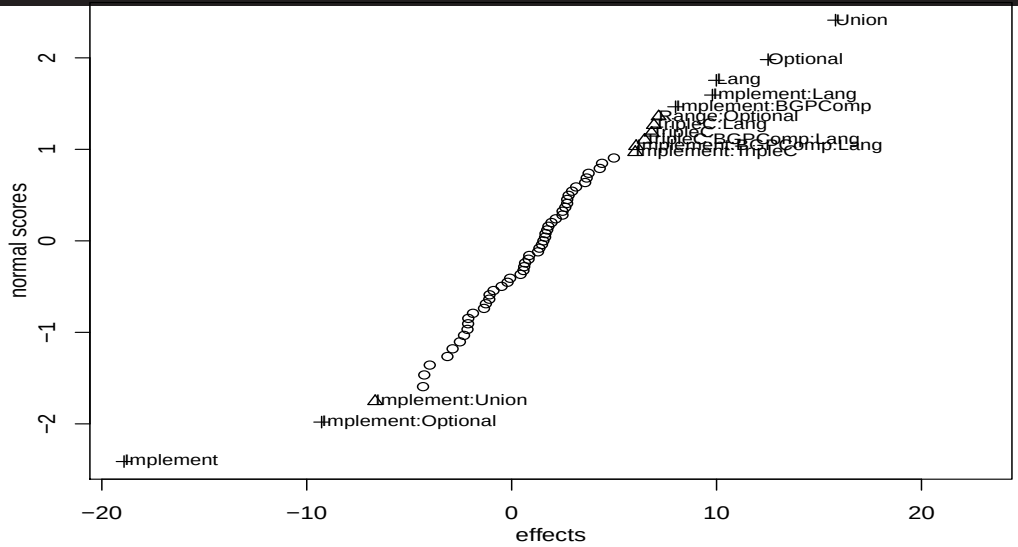
Experimental Economy

- Full factorial experimental goes as 2^n.
- SPARQL Endpoint evaluation is inherently complex, with many possible factors.
- May reduce the size of the experiment with Fractional Factorial Experiments.
- The cost is explanatory power due to *aliasing*.
- E.g. we cannot tell the difference of an effect of “TripleC::BGPComp” from “Machine::Range”
- Easy in R to declare which effects must not be aliased.
- We have made two designs: One 32-run and one 64-run.

32-run experiment



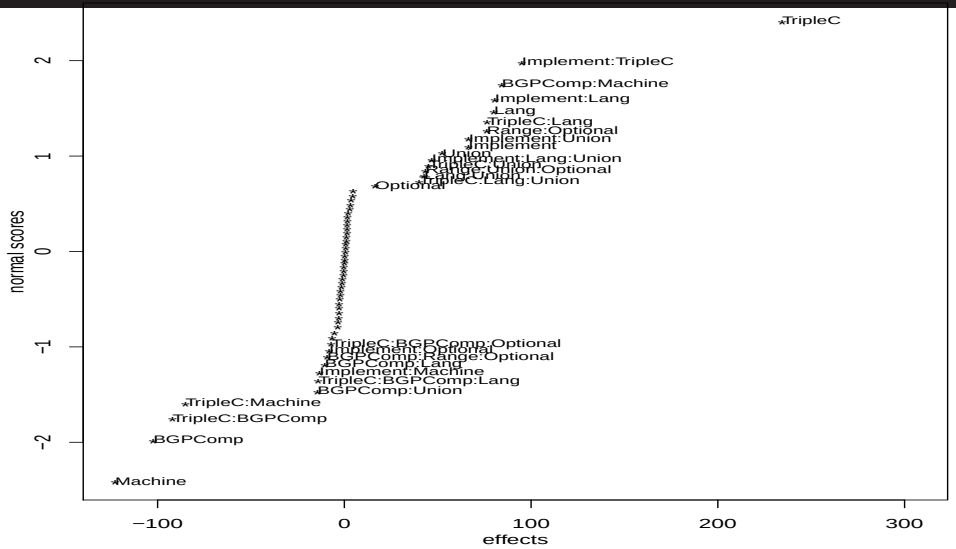
64-run experiment



Analysis of 64-run experiment

- Find most of the same significant effects as the full experiment.
- However, we see that “Implement:TripleC” emerges only for $\alpha = 0.15$.
- “Machine”, “TripleC” and “Range” are inactive, we can perform hypothesis test as above, and find $p = 5.9 \cdot 10^{-6}$.

Fractional experiment with more data



The role of randomization

- We have allowed us to disregard the effect of caching, warm-up-runs, etc.
- This is different from neutralizing the effect!
- They are now “lurking variables”.
- They contribute to the overall unexplained variance.
- Unexplained variance should be kept to a minimum.
- Time-dependent or order-dependent factors may be required.

Choice of factors

- Different strategies must be tried, open field.
- Parameterization is very important.
  - Data heterogeneity, skewed distributions, etc.
  - Parameterizing SPARQL queries, see e.g. SPLODGE by Görlitz et al.
  - Parameterizing SPARQL queries using the grammar pragmatically?

Other important issues

- Comparing completely different SPARQL implementations.
- Ensuring that small and efficient experiments provide enough details to assess the soundness of the experiment.
- This paper scratches the surface of DoE, Orthogonal Arrays is a more general formalism.
- This topic is also interesting for progress beyond the state-of-the-art in statistics.

We saw

- ... an experimental setup using DoE.
- ... how the analysis pointed out the important effects.
- ... how we got a comprehensive view of the experiment.
- ... how hypothesis tests could be formulated.
- ... how 2-level experiments can determine significant effects, even though it is not good enough for modelling.
- ... how more economic experiments can be designed, and noted their limitations.
- ... finally how the experiment could be shown to be flawed.

Questions to ask

1. Are there factors that cover all realistic features?
2. If not, are they adequately covered by randomization?
3. If so, would the variance resulting from randomization obscure factors that could provide clues that the levels are wrongly set?
4. By carefully examining interactions with “Implement”, are there any that are unaccounted for, and that could point out wrongly set levels?

What's a worthy result?

- I have not made this a primary focus of my work.
- Even though I think I should.
- 3000 full-time researchers to construct the ATLAS experiment at CERN.
- Improvement in experimental methodology should be a worthy result.

Thank you!

- Code and instructions at Github: `https://github.com/kjetilk/doe-sparql`.
- Happy to help getting it to run!
- Happy to discuss collaborations or workshops on this topic!