Driected Random

Abdullah Alsharif

3 January 2017

Test generation Timing Graph

Figure 1 shows the average test generation timing for each technique for each DBMS, for all runs and schemas. Just By looking at the graph it shows that Directed Random is much faster compared to AVM in generating tests nearly 1 second faster.

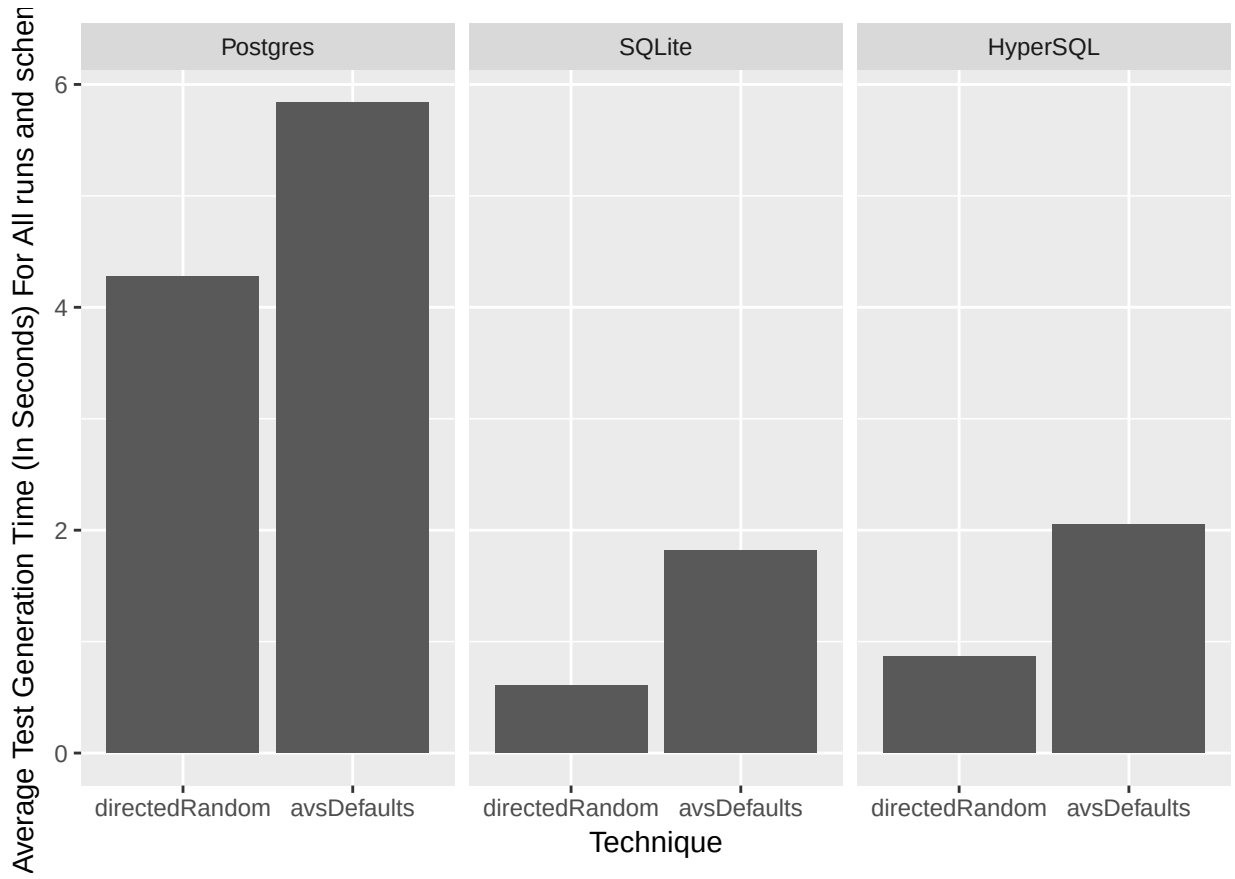

Figure 1: Avrages of Test generation timing - in seconds

In Figure 2 we review average test generation timing for each techinque for each schema split by DBMSs and for all runs. We can see that Directed Random still winning for each schema (NOTE we are still waiting for iTrust data).

In Figure 3 I show the spread of values of test generation times in regard of DBMS and technique using a box plot, for all runs and schemas.

In Figure 11 I show the spread of values for test generation timing for each schema, DBMS and technique, for all runs.

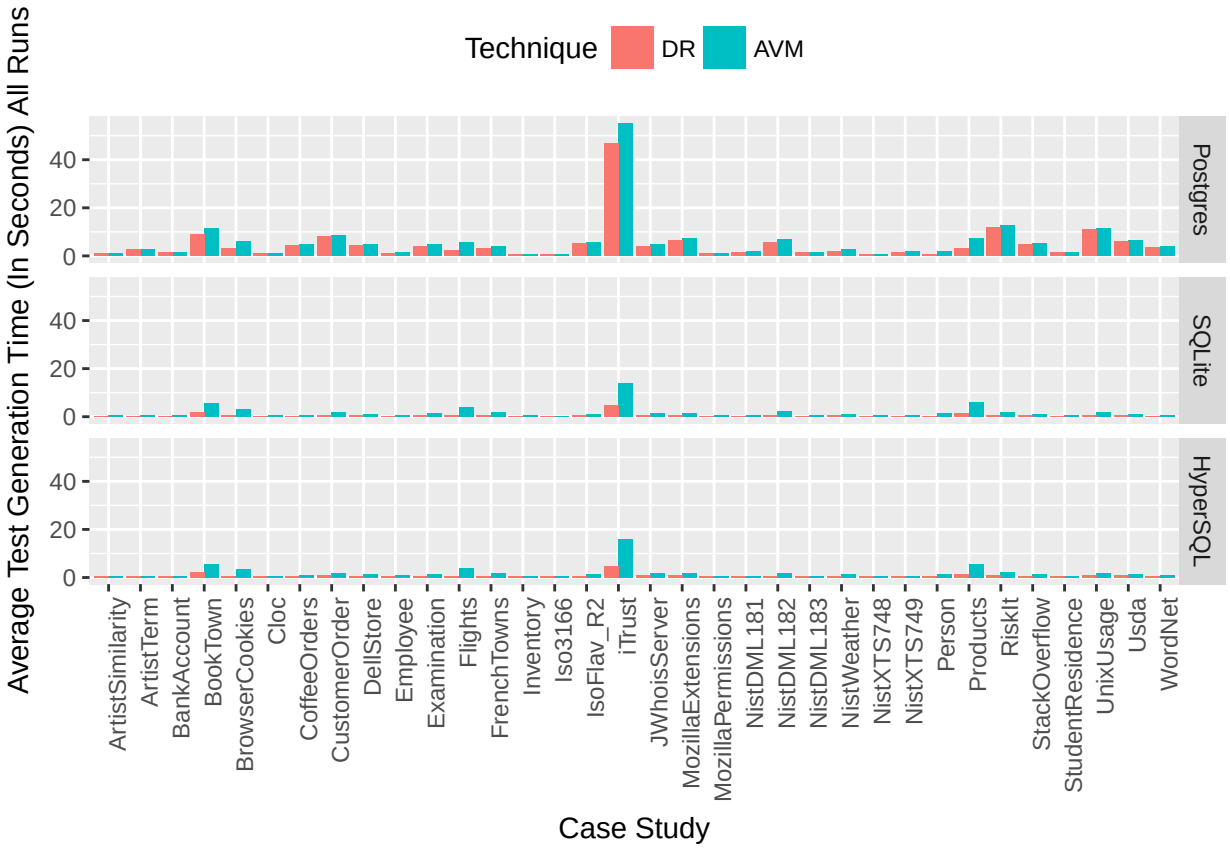

Figure 2: Avrages of Test generation timing for each schema- in seconds

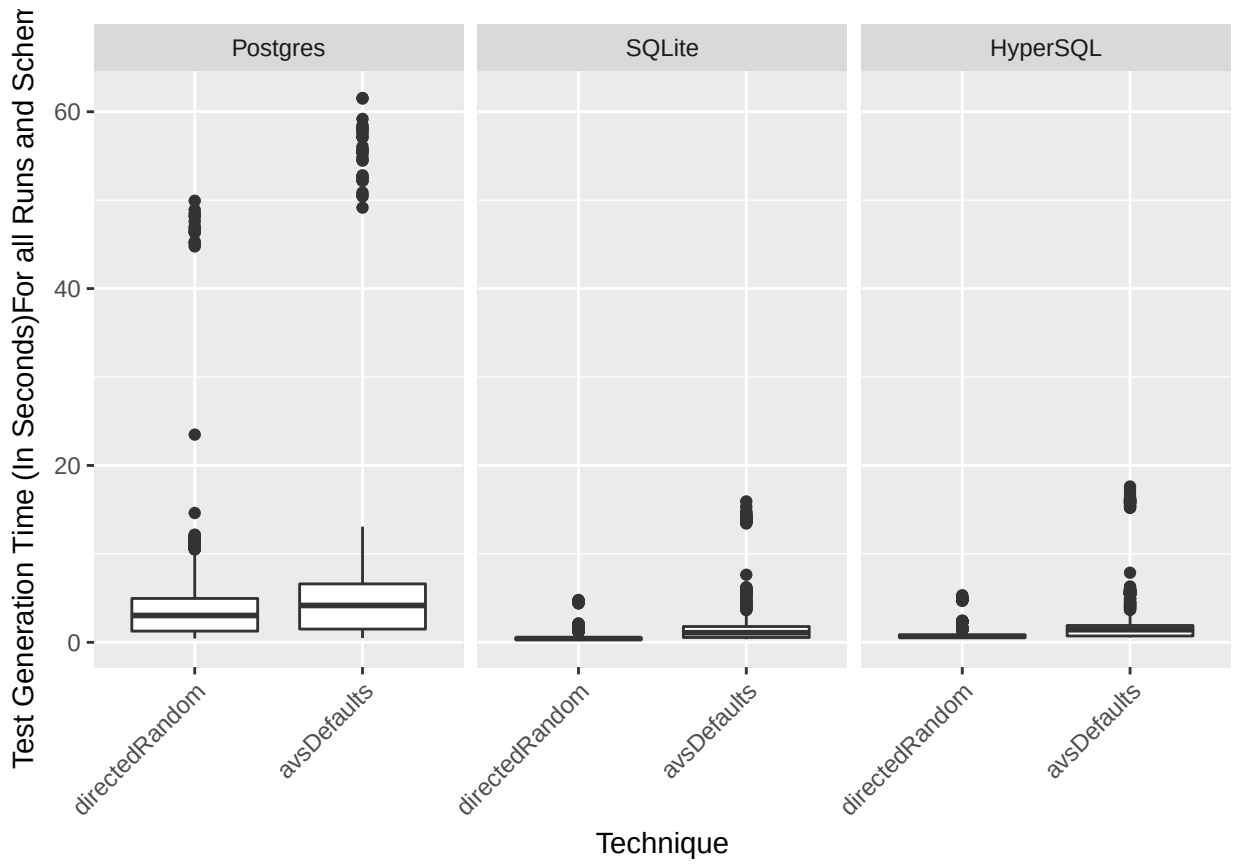

Figure 3: Test generation timing - in seconds

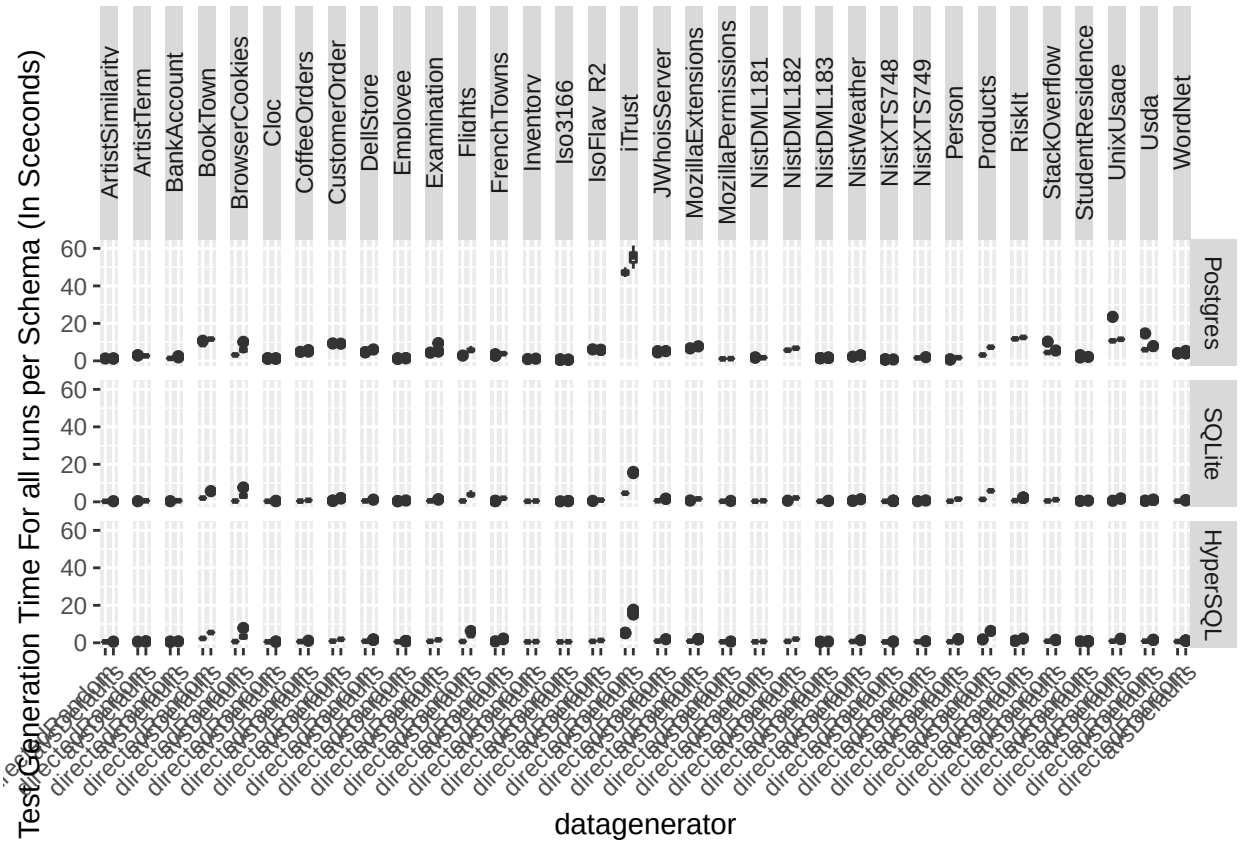

Figure 4: Box plot for mutation scores for DBMS, techniques and schemas - in precentage

Mutation Scores

In Figure 4 I shows the average mutation score for each technique for each DBMS, for all runs and schemas. Just By looking at the graph it shows that Directed Random is batter when compared to AVM in killing more mutants.

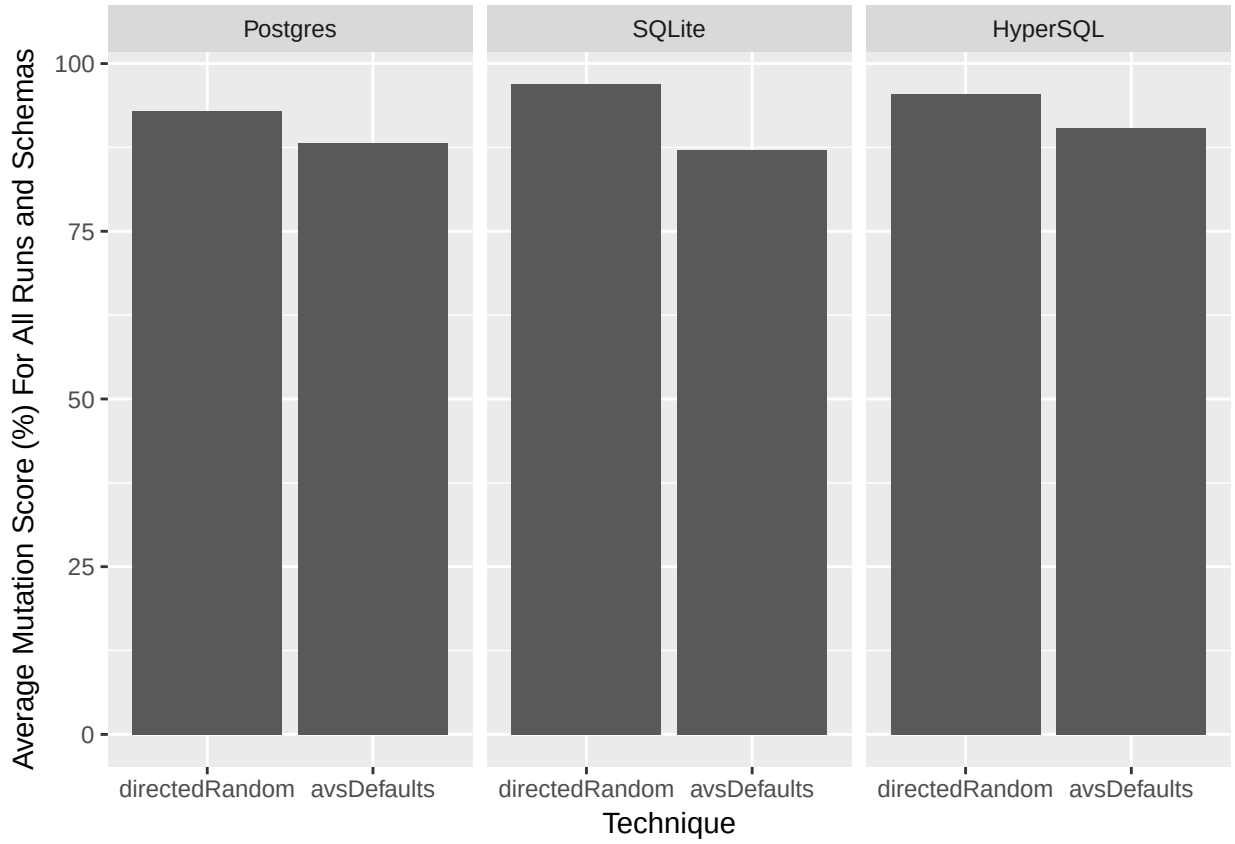

Figure 5: Avrages of Mutation Score - in precentage

In Figure 5 I review average mutation score for each techinque for each schema split by DBMSs, for all runs. We can see that Directed Random have a better or similar scores to AVM however not even one schema has less score comparing to AVM (NOTE we are still waiting for iTrust data).

In Figure 6 I show the spread of values of mutation score in regard of DBMS and technique using a box plot, for all runs and schemas.

In Figure 7 I show the spread of values for mutation scores for each schema, DBMS and technique, for all runs.

Mutation Scores and Mutation Operators

In Figure 8 I shows the average mutation score for each technique for each DBMS and the mutatan operators, for all runs, schemas and not including Equvilant, Redundant Quasi mutants . Just By looking at the graph it shows that Directed Random is batter when compared to AVM in killing more mutants for two operators the rest shows same percentages.

In Figure 9 I shows a box plot mutation score for each technique for each DBMS and the mutatan operators, this will help us see the spread of values. Just By looking at the graph it shows that Directed Random is batter when compared to AVM in killing more mutants for two operators the rest shows same percentages.

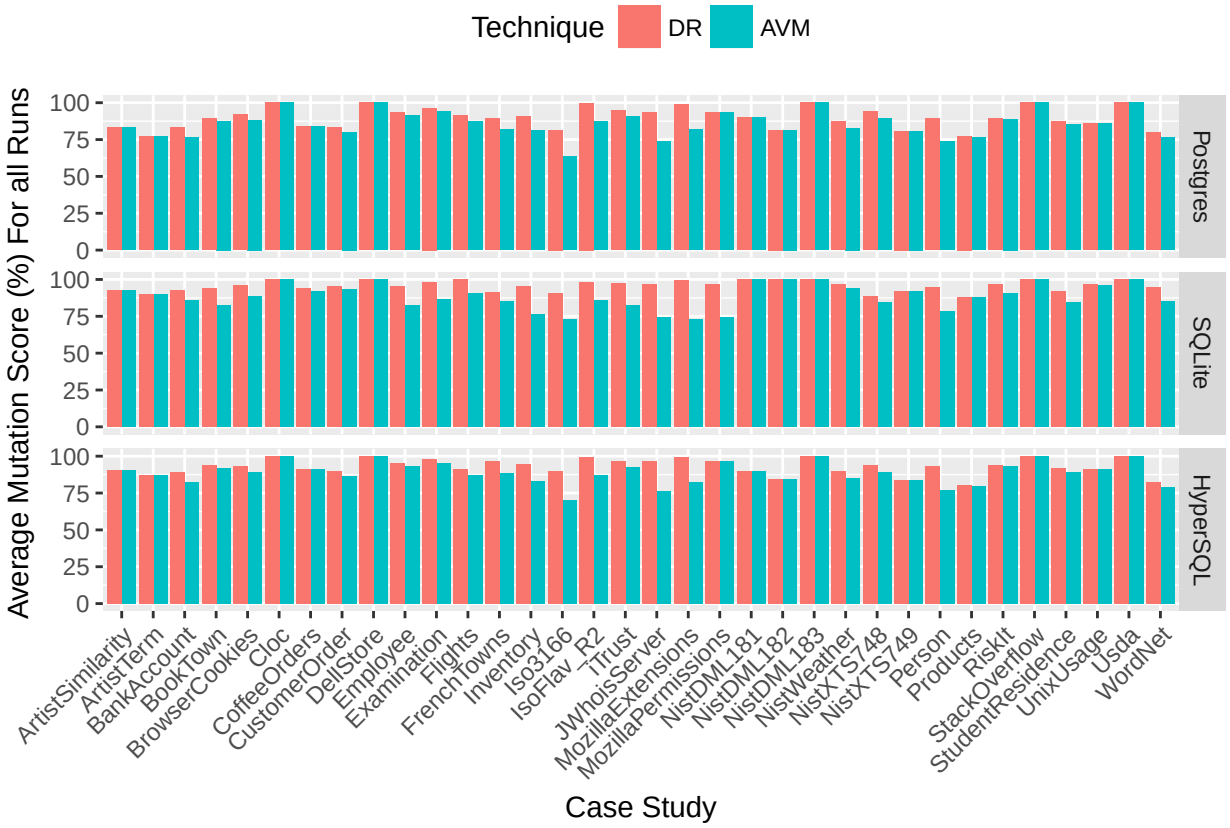

Figure 6: Avrages of Mutation Score for each schema - in precentage

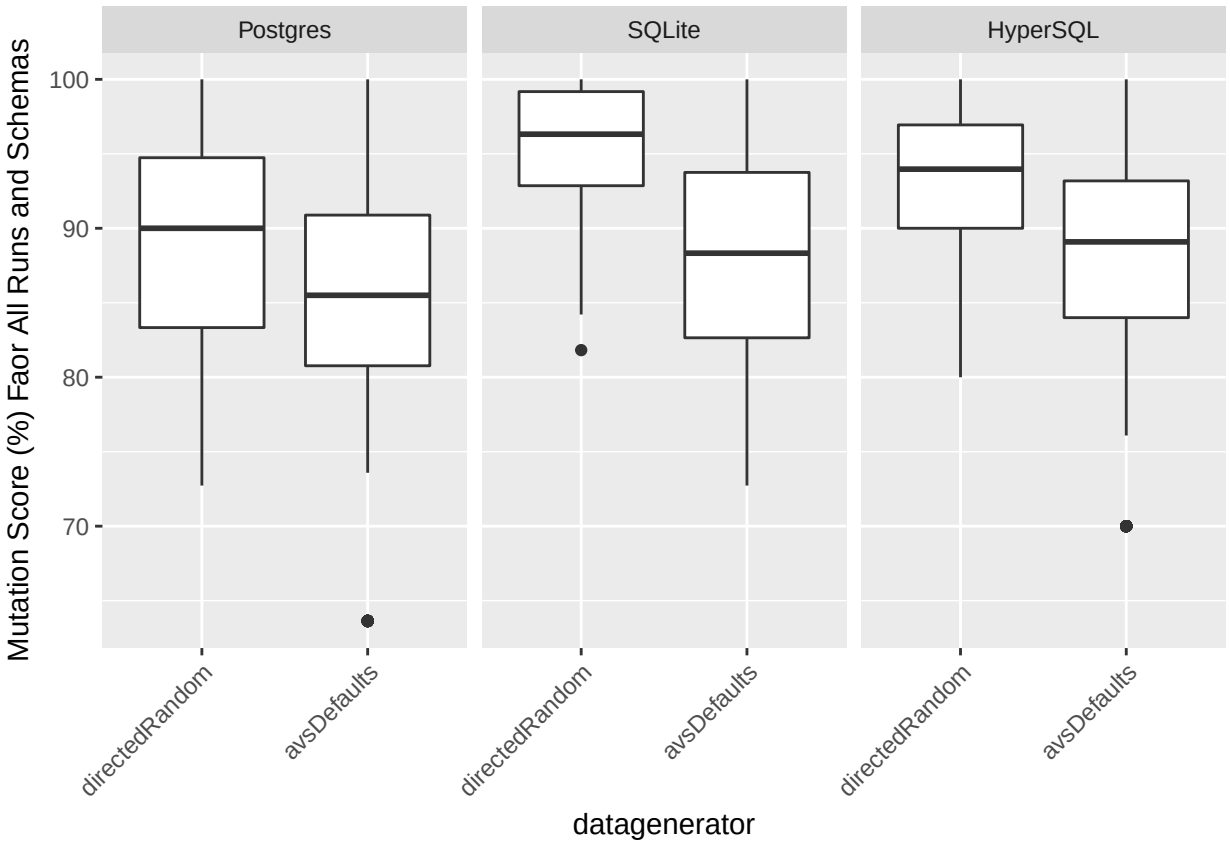

Figure 7: Box plot for mutation scores for DBMSs and techniques - in precentage

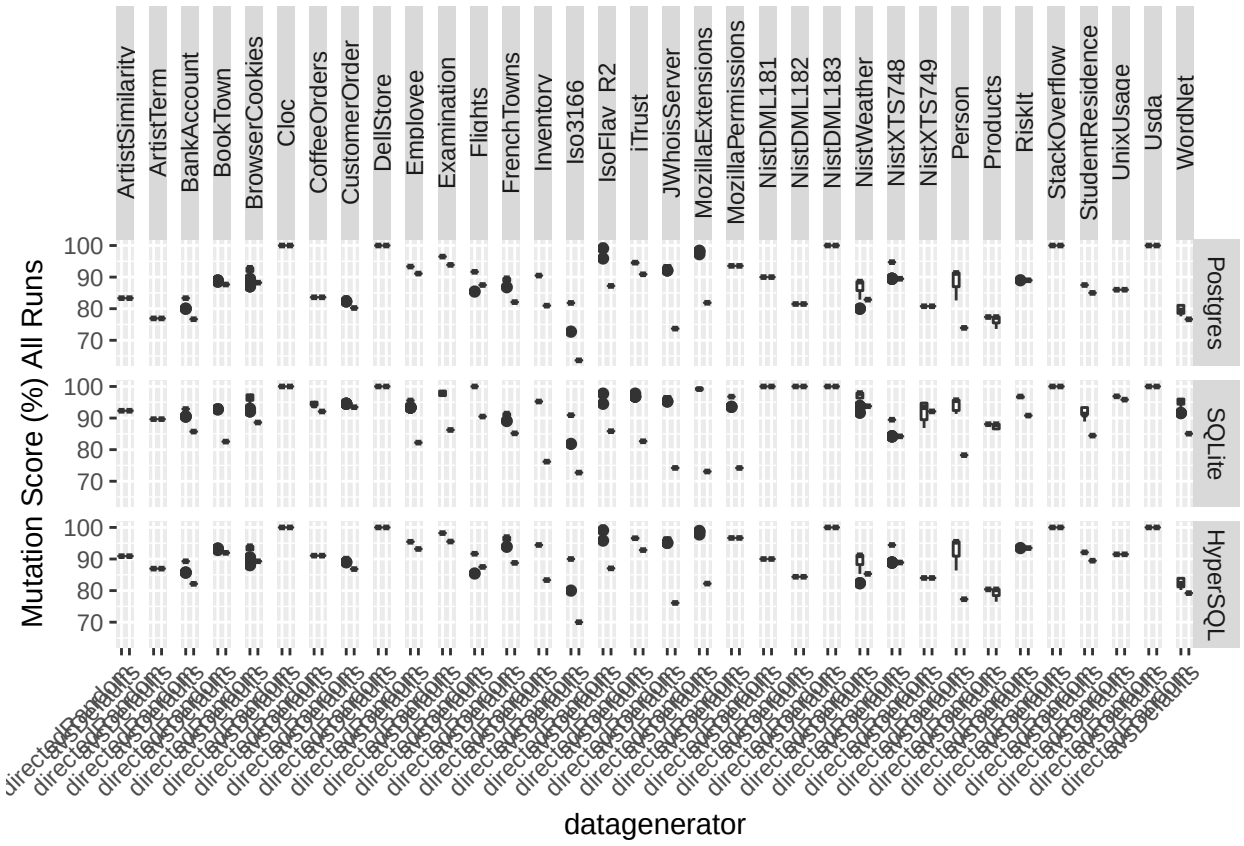

Figure 8: Box plot for mutation scores for DBMS, techniques and schemas - in precentage

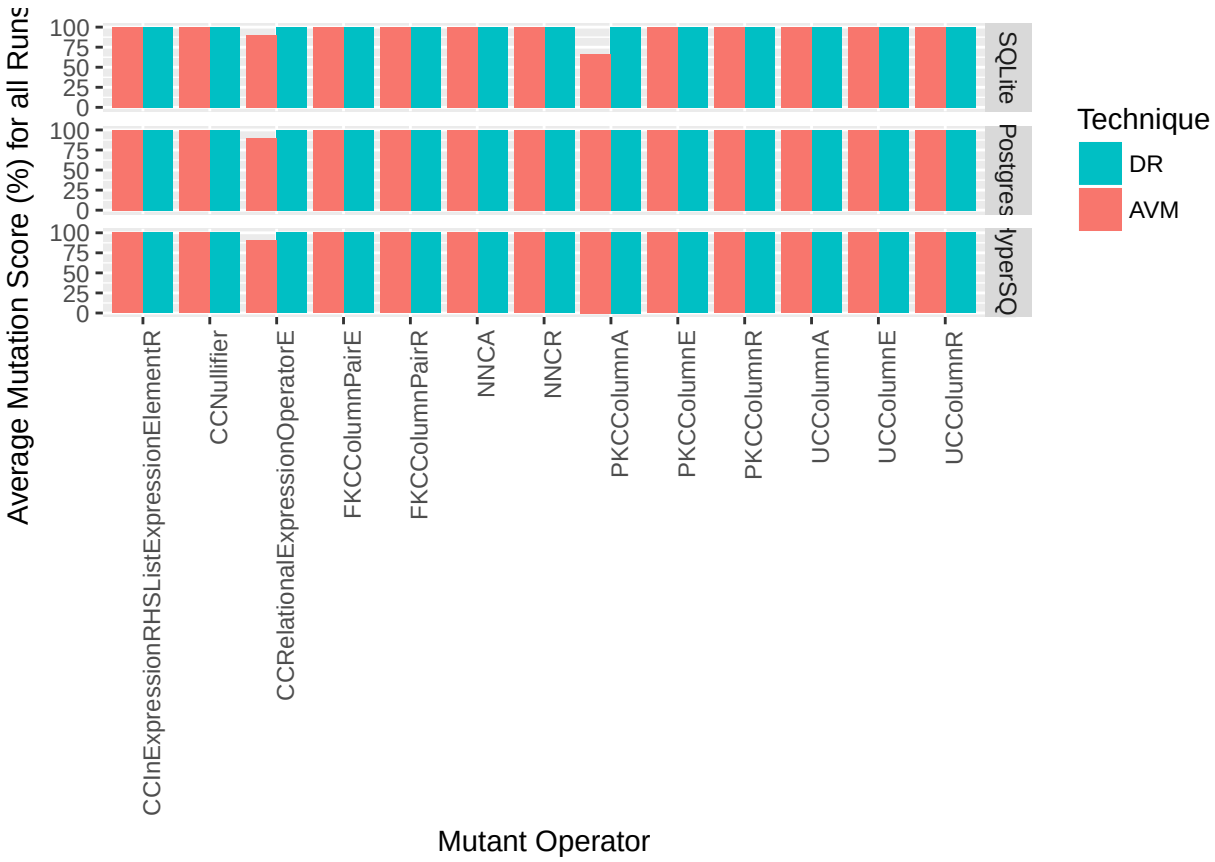

Figure 9: Averages of mutant scores in regard of Mutant Operators and DBMSs - in precentage

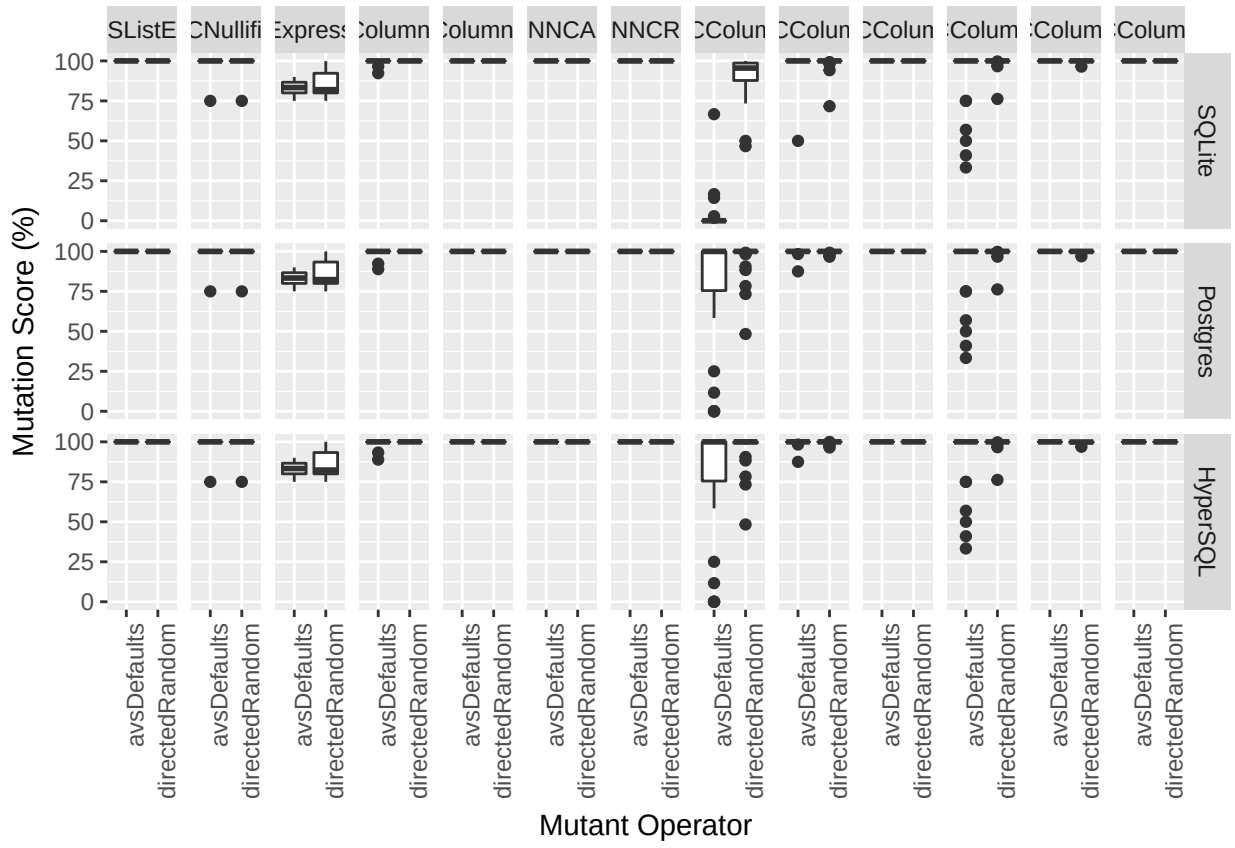

Figure 10: Box Plot of mutant scores in regard of Mutant Operators and DBMSs - in precentage

Analyse Wilcox Rank and Effect Size for AVM and Directed Random

To statistically analyze the the new technique we conducted tests for significance with the nonparametric Wilcoxon rank-sum test, using the sets of 30 execution times obtained with a specific DBMS and all techniques **Hitchhicker Guide Ref**. A p-value that less than 0.05 is considered significant. To review practical are significance tests we use the nonparametric A12 statistic of Vargha and Delaney **REF** was used to calculate effect sizes. The A12 determine the average probability that one approach beats another, or how superior one technique compared to the other. We followed the guidelines of Vargha and Delaney in that an effect size is considered to be "large" if the value of A12 is $ < 0:29$ or > 0.71, "medium" if A12 is < 0.36 or > 0.64 and "small" if A12 is < 0.44 or > 0.56. However, is the values of A12 close to the 0.5 value are viewed as there no effect.

When comparing AVM and Directed Random techniques in regard of time we used Mann-Whitney U-test and the A-hat effect size calculations. As Shown in in the following two tables that there is statistically significat differance between Driected Random and AVM, $p \leq 0.05$. Therefore, we reject the null hypothesis that there is no difference between AVM and Directed Random. As p-value near zero, that directed random is faster than AVM in a statistically significant test. Moreover, the A-12 shows that Directed Random has a large effect size when it comes to test generation timing. Which menas that Directed Random is the winner in regard of test generation timing.

p.value	dbms	vs
0.93791	Postgres	AVM vs Directed Random Coverage
1.00000	SQLite	AVM vs Directed Random Coverage
1.00000	HyperSQL	AVM vs Directed Random Coverage
0.00000	Postgres	AVM vs Directed Random Mutation Score
0.00000	SQLite	AVM vs Directed Random Mutation Score
0.00000	HyperSQL	AVM vs Directed Random Mutation Score
0.00000	Postgres	AVM vs Directed Random Number Of evaluations
0.00000	SQLite	AVM vs Directed Random Number Of evaluations
0.00000	HyperSQL	AVM vs Directed Random Number Of evaluations
0.00000	Postgres	AVM vs Directed Random Test Generation Time
0.00000	SQLite	AVM vs Directed Random Test Generation Time
0.00000	HyperSQL	AVM vs Directed Random Test Generation Time

size	rank.sum	dbms	threshold
large	465	Postgres	0
large	465	SQLite	0
large	465	HyperSQL	0
large	465	Postgres	100
large	465	SQLite	100
large	465	HyperSQL	100
large	465	Postgres	200
large	465	SQLite	200
large	465	HyperSQL	200
large	465	Postgres	300
large	465	SQLite	300
large	465	HyperSQL	300
large	465	Postgres	400
large	465	SQLite	400
large	465	HyperSQL	400
large	465	Postgres	500
large	465	SQLite	500

size	rank.sum	dbms	threshold
large	465	HyperSQL	500
large	465	Postgres	600
large	465	SQLite	600
large	465	HyperSQL	600
large	465	Postgres	700
large	465	SQLite	700
large	465	HyperSQL	700
large	465	Postgres	800
large	465	SQLite	800
large	465	HyperSQL	800
large	465	Postgres	900
large	465	SQLite	900
large	465	HyperSQL	900
large	465	Postgres	1000
large	465	SQLite	1000
large	465	HyperSQL	1000

HeatMap Plot of mutant analysis per technique for each Operator

Some Text Description for the heat map

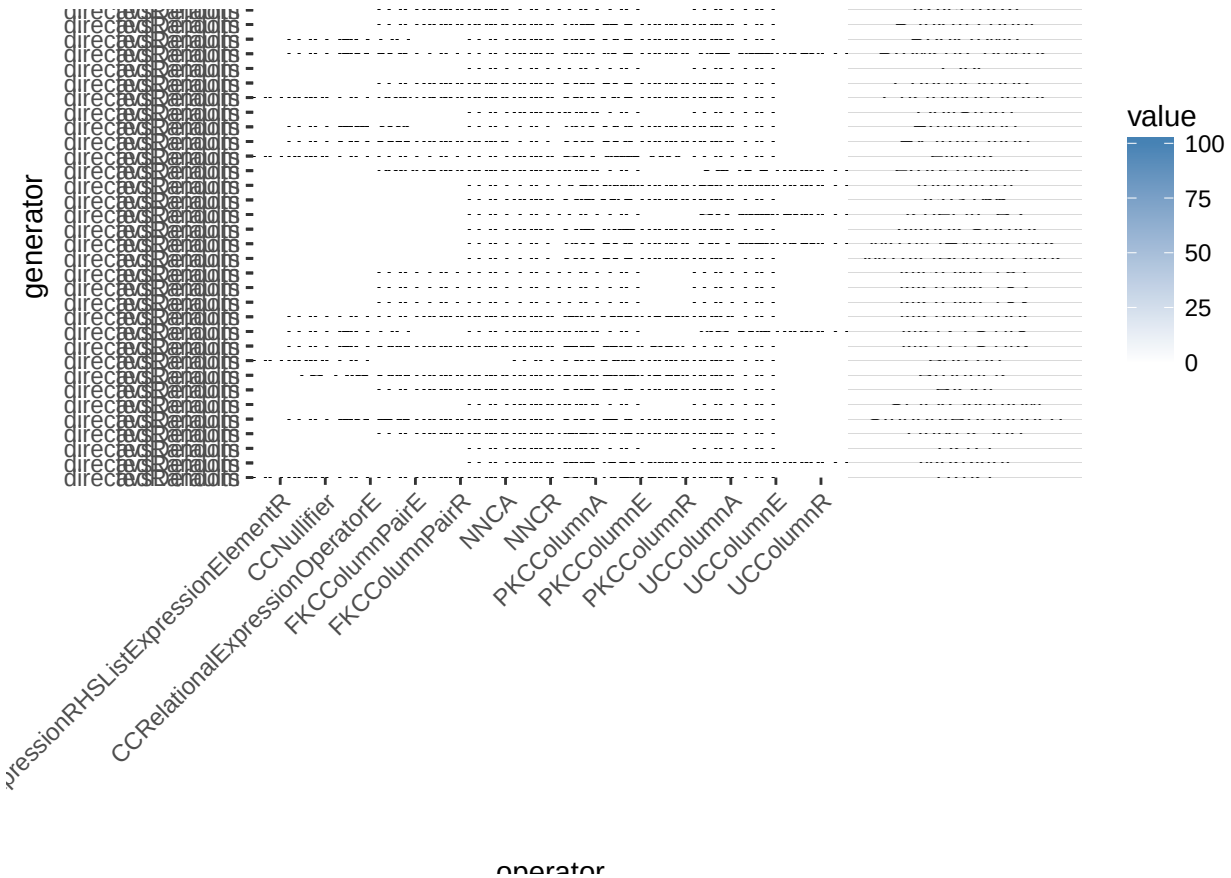

Figure 11: Heat Map of mutant scores in regard of Mutant Operators and DBMSs - in precentage