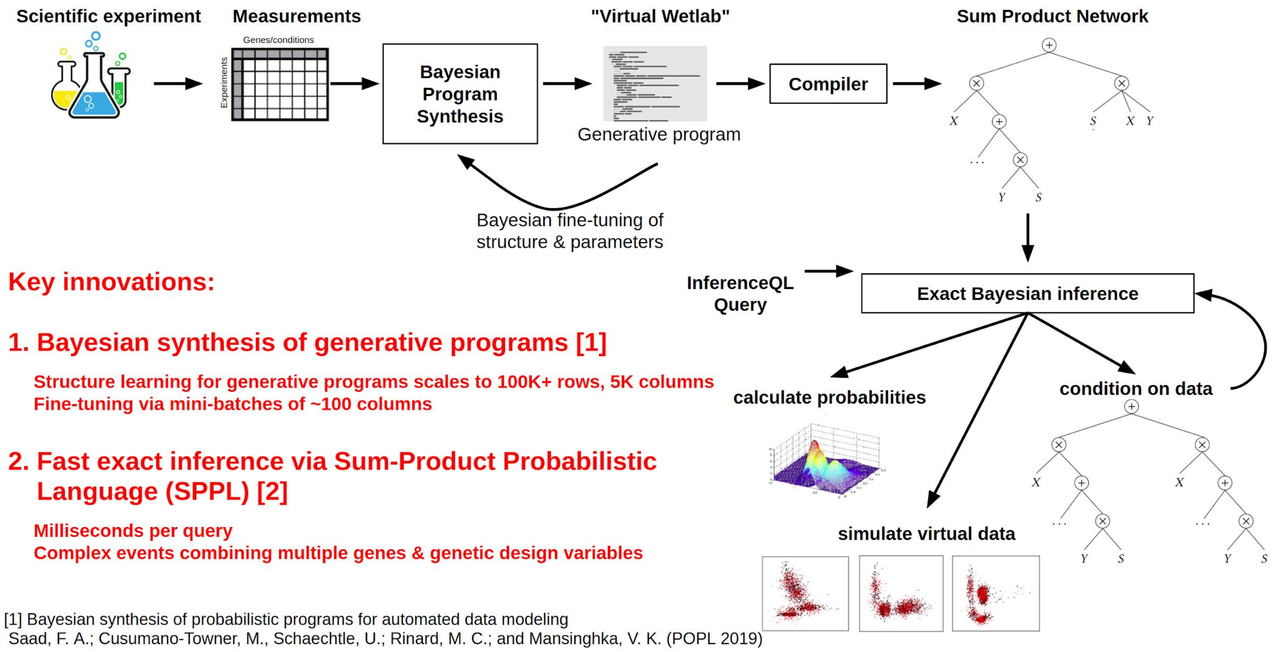
InferenceQL: an SQL-like probabilistic programming language

Ulrich Schaechtle Zane Shelby Vikash K. Mansinghka

MIT Probabilistic Computing Project

1. InferenceQL platform



Key innovations:

1. Bayesian synthesis of generative programs [1]

- Structure learning for generative programs scales to 100K+ rows, 5K columns
- Fine-tuning via mini-batches of ~100 columns

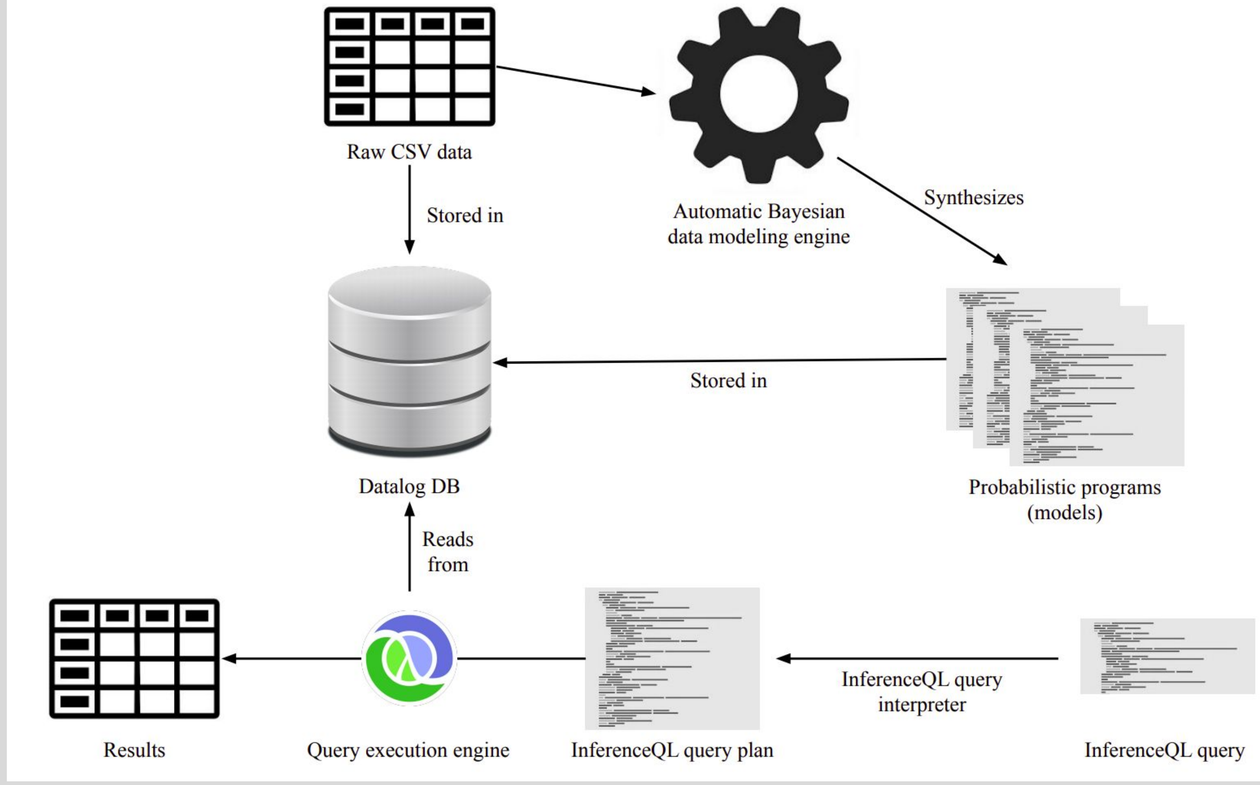
2. Fast exact inference via Sum-Product Probabilistic Language (SPPL) [2]

- Milliseconds per query
- Complex events combining multiple genes & genetic design variables

[1] Bayesian synthesis of probabilistic programs for automated data modeling
Saad, F. A.; Cusumano-Towner, M., Schaechtle, U.; Rinard, M. C.; and Mansinghka, V. K. (POPL 2019)

[2] Exact Symbolic Inference in Probabilistic Programs via Sum-Product Representations.
Saad, F. A.; Rinard, M. C.; and Mansinghka, V. K. (in review for POPL 2020)

2. InferenceQL system architecture



3. Example InferenceQL application: genetic circuit design

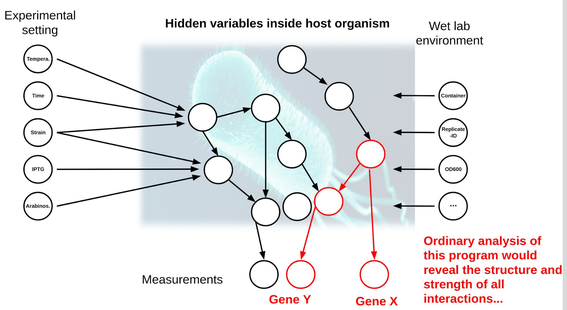
Experimental data available for genetic circuit design

Experimental condition | Circuit parts | ~ 4095 Genes that are not part of the circuit

~ 945 rows Measurements (FPKM)

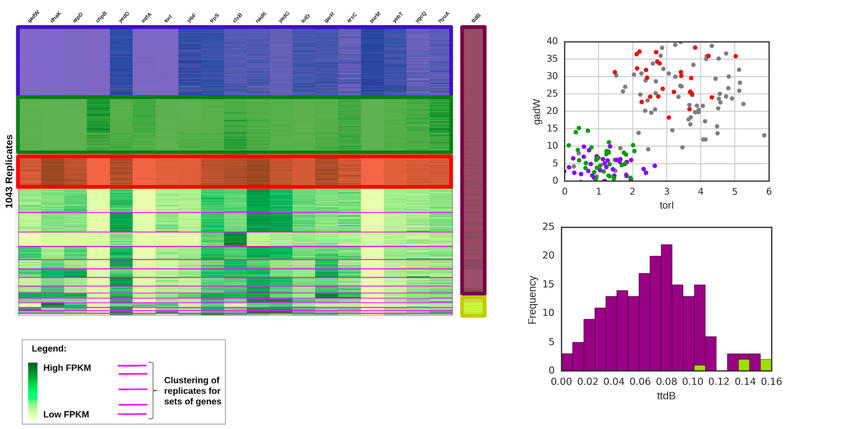
Arabinose	IPTG	Part_KanR	Part_AraC	Part_LacI	Part_AmeRv2	Part_BM3R1	Part_PhlF	Part_PsrA	Part_YFP	...	ybgF	rpoC	ychJ
0	1	NaN	72.788976	1.879291	NaN	NaN	NaN	NaN	8.614201	...	7.915932	20.797303	2.087655
1	0	NaN	NaN	1.584695	NaN	NaN	NaN	42.755966	688.682559	...	8.045223	21.904313	2.857201
0	1	NaN	NaN	3.157314	143.937934	NaN	NaN	NaN	6.228926	...	15.409489	42.510479	0.412231
0	1	NaN	80.142359	1.678059	NaN	NaN	NaN	NaN	6.180037	...	7.332865	23.701846	1.670575
0	0	NaN	42.842255	3.745004	NaN	NaN	NaN	NaN	23.559391	...	13.872782	36.121186	0.910976
1	1	NaN	NaN	0.783051	NaN	NaN	NaN	812.772487	3.891203	...	7.734214	25.318934	4.655556
0	1	NaN	61.734561	2.110796	NaN	NaN	12.207818	NaN	1891.274032	...	7.193015	20.096298	3.177637
0	0	NaN	50.449965	3.503470	15.998515	NaN	12.915777	NaN	1344.816606	...	13.638680	31.452056	1.107454
0	0	NaN	75.565985	1.929219	NaN	25.753129	11.237290	NaN	NaN	...	10.024819	31.644259	2.225864
0	1	NaN	40.281101	3.281829	NaN	574.900627	NaN	NaN	NaN	...	13.693014	37.007891	0.714531
0	0	NaN	47.590196	3.775271	17.988969	NaN	NaN	NaN	NaN	...	12.803708	34.764955	0.789024

What if we had a program that simulated virtual experiments?



Bayesian program synthesis produces a generative program that can serve as a "virtual wetlab"

```
(define simulator
  (gen []
    (define cluster-id-for-torI-yibF
      (categorical [0.249252 0.204387 0.110668 [...]]))
    (define [torI-mean torI-std]
      (cond
        (= cluster-id-for-torI-yibF 0) [1.072683 0.435928]
        (= cluster-id-for-torI-yibF 1) [0.920622 0.394361]
        (= cluster-id-for-torI-yibF 2) [3.280830 0.814852]
        [...]))
    (define torI
      (gaussian torI-mean torI-std))
    (define [gadW-mean gadW-std]
      (cond
        (= cluster-id-for-torI-gadW 1) [4.522248  1.208316]
        (= cluster-id-for-torI-gadW 2) [5.862314  3.266713]
        (= cluster-id-for-torI-gadW 0) [30.837259 7.603366]
      [...])))
    (define gadW
      (gaussian gadW-mean gadW-std))
    (define cluster-id-for-ttdB
      (categorical [0.936232 0.063768]))
      (define [ttdB-mean ttdB-std]
        (cond
          (= cluster-id-for-ttdB 0) [0.064825 0.035780]
          (= cluster-id-for-ttdB 1) [0.158588 0.036251]))
      (define ttdB
        (gaussian ttdB-mean ttdB-std))
    [torI gadW ttdB]))
```



4. Example query and accuracy results: predicting more than 4000 genes accurately

SQL-like probabilistic query

```
SELECT
  MEDIAN(aaeA),
  MEDIAN(aaeB),
  [...],
  MEDIAN(zur),
  MEDIAN(zwf)
FROM (GENERATE aaeA, aaeB, [...], zur, zwf
    GIVEN
      Arabinose = 'on' AND IPTG = 'on' AND  Measurement_phase = 'log-phase' AND
      68.29/2  < Part_AraC AND Part_AraC <   68.29 * 2 AND
      1.77/2   < Part_LacI AND Part_LacI <    1.77 * 2 AND
      828.80/2 < Part_AmeR AND Part_AmeR <  828.80 * 2 AND
      603.22/2 < Part_PhlF AND Part_PhlF <  603.22 * 2 AND
      1.57/2   < Part_YFP  AND Part_YFP  <    1.57 * 2
    UNDER whole_genome_spn LIMIT 1000)
```

We know these values a priori:

Train / test data split

Training data: single parts / "gates"

- 4095 columns,
- 945 rows

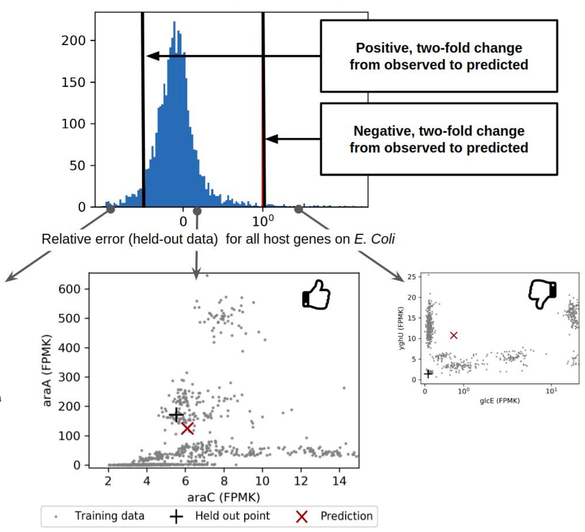
araC (sensor) — pBAD-PsrA
lacI (sensor) — pBAD-PhlF
pTac-BM3R1

Testing data: complex circuits

- 4095 columns,
- 96 rows

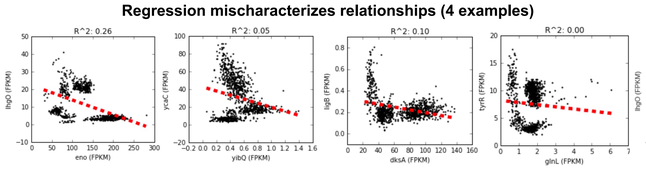
pBAD-PhlF, pTac-AmeR → YFP
pBAD-PhlF, pTac-BM3R1 → YFP
pBAD-PsrA, pTac-AmeR → YFP

Accuracy results: Only 6.51% of predicted gene expression have a fold change error larger than 2x



Qualitative assessment: InferenceQL models the dynamics in the data more accurately than stand approaches

Regression mischaracterizes relationships (4 examples)



Virtual data from the whole-genome simulator matches real data