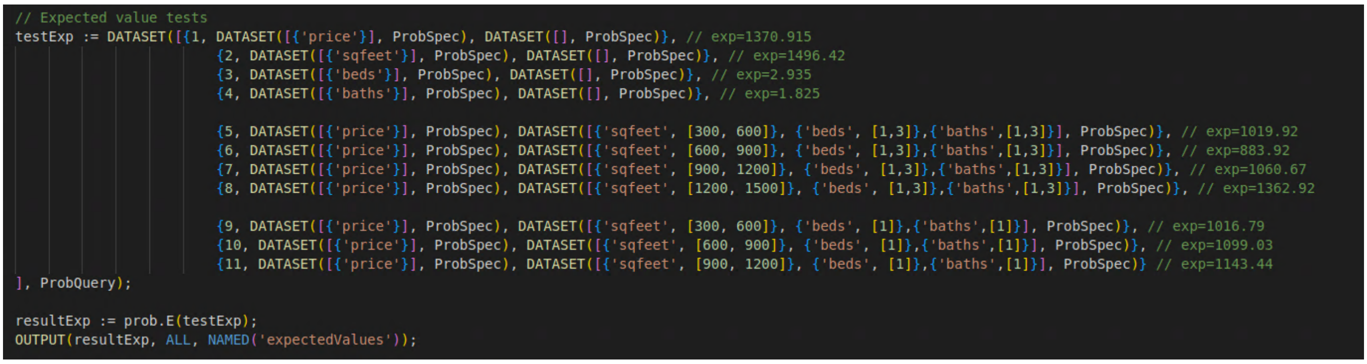
HOME | Timeline | Previous Week | Next Week

Monday 06/20

A fantastic morning to start with. In a meeting with Roger and Gordon, we discussed how to visualize the graphs. We have also discussed Observable JavaScript and ECL notebook and visualizing graphs there. After that meeting, Roger and I have decided to, for the time being, let's focus on the only type of house in the 'housing' dataset and continue the analysis. To convert the 'Type' of the housing dataset to the numeric data format, Roger suggested me use LabelEncoder from ML_Core. I have reported the error that I have faced in 'ContinuousTest.ecl' of the HPCC_Causality bundle.

I have started analyzing the probability distribution and dependency of variables in the 'housing' dataset. I have found that 75% of the *house*s have a price in the range of $500-$1700. And only 4% of the houses lie in the range from $3000-$10000. Some of the probability observations have been made.

Expectation Value testing:



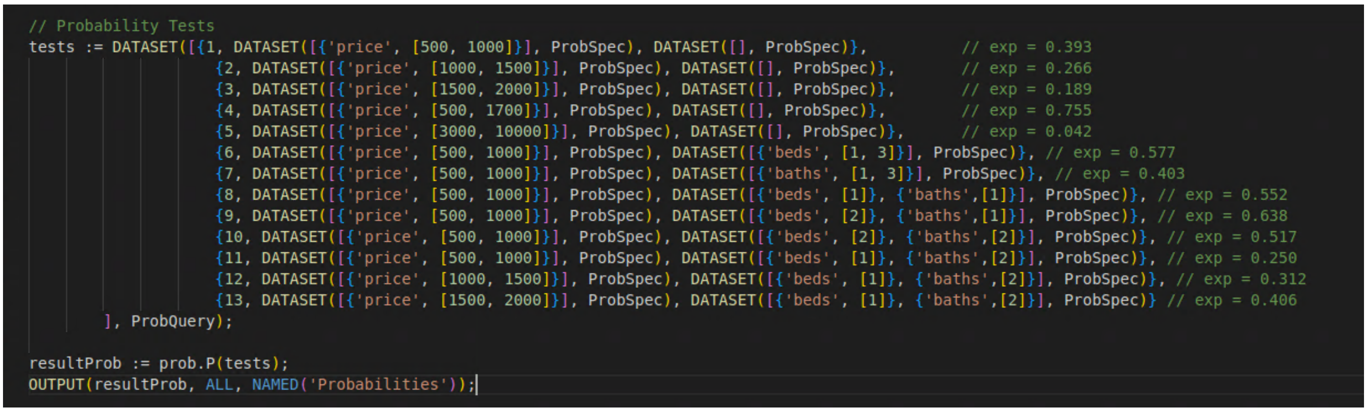
```
// Expected value tests
testExp := DATASET([{1, DATASET([{'price'}], ProbSpec), DATASET([], ProbSpec)}, // exp=1370.915
                    {2, DATASET([{'sqfeet'}], ProbSpec), DATASET([], ProbSpec)}, // exp=1496.42
                    {3, DATASET([{'beds'}], ProbSpec), DATASET([], ProbSpec)}, // exp=2.935
                    {4, DATASET([{'baths'}], ProbSpec), DATASET([], ProbSpec)}, // exp=1.825

                    {5, DATASET([{'price'}], ProbSpec), DATASET([{'sqfeet', [300, 600]}, {'beds', [1,3]},{'baths',[1,3]}], ProbSpec)}, // exp=1019.92
                    {6, DATASET([{'price'}], ProbSpec), DATASET([{'sqfeet', [600, 900]}, {'beds', [1,3]},{'baths',[1,3]}], ProbSpec)}, // exp=883.92
                    {7, DATASET([{'price'}], ProbSpec), DATASET([{'sqfeet', [900, 1200]}, {'beds', [1,3]},{'baths',[1,3]}], ProbSpec)}, // exp=1060.67
                    {8, DATASET([{'price'}], ProbSpec), DATASET([{'sqfeet', [1200, 1500]}, {'beds', [1,3]},{'baths',[1,3]}], ProbSpec)}, // exp=1362.92

                    {9, DATASET([{'price'}], ProbSpec), DATASET([{'sqfeet', [300, 600]}, {'beds', [1]},{'baths',[1]}], ProbSpec)}, // exp=1016.79
                    {10, DATASET([{'price'}], ProbSpec), DATASET([{'sqfeet', [600, 900]}, {'beds', [1]},{'baths',[1]}], ProbSpec)}, // exp=1099.03
                    {11, DATASET([{'price'}], ProbSpec), DATASET([{'sqfeet', [900, 1200]}, {'beds', [1]},{'baths',[1]}], ProbSpec)} // exp=1143.44
], ProbQuery);

resultExp := prob.E(testExp);
OUTPUT(resultExp, ALL, NAMED('expectedValues'));
```

Some Probability and conditional probability observations have been made.



```
 // Probability Tests
 tests := DATASET([{1, DATASET([{'price', [500, 1000]}], ProbSpec), DATASET([], ProbSpec)},          // exp = 0.393
                   {2, DATASET([{'price', [1000, 1500]}], ProbSpec), DATASET([], ProbSpec)},       // exp = 0.266
                   {3, DATASET([{'price', [1500, 2000]}], ProbSpec), DATASET([], ProbSpec)},       // exp = 0.189
                   {4, DATASET([{'price', [500, 1700]}], ProbSpec), DATASET([], ProbSpec)},        // exp = 0.755
                   {5, DATASET([{'price', [3000, 10000]}], ProbSpec), DATASET([], ProbSpec)},      // exp = 0.042
                   {6, DATASET([{'price', [500, 1000]}], ProbSpec), DATASET([{'beds', [1, 3]}], ProbSpec)}, // exp = 0.577
                   {7, DATASET([{'price', [500, 1000]}], ProbSpec), DATASET([{'baths', [1, 3]}], ProbSpec)}, // exp = 0.403
                   {8, DATASET([{'price', [500, 1000]}], ProbSpec), DATASET([{'beds', [1]}, {'baths',[1]}], ProbSpec)}, // exp = 0.552
                   {9, DATASET([{'price', [500, 1000]}], ProbSpec), DATASET([{'beds', [2]}, {'baths',[1]}], ProbSpec)}, // exp = 0.638
                   {10, DATASET([{'price', [500, 1000]}], ProbSpec), DATASET([{'beds', [2]}, {'baths',[2]}], ProbSpec)}, // exp = 0.517
                   {11, DATASET([{'price', [500, 1000]}], ProbSpec), DATASET([{'beds', [1]}, {'baths',[2]}], ProbSpec)}, // exp = 0.250
                   {12, DATASET([{'price', [1000, 1500]}], ProbSpec), DATASET([{'beds', [1]}, {'baths',[2]}], ProbSpec)}, // exp = 0.312
                   {13, DATASET([{'price', [1500, 2000]}], ProbSpec), DATASET([{'beds', [1]}, {'baths',[2]}], ProbSpec)} // exp = 0.406
         ], ProbQuery);

 resultProb := prob.P(tests);
 OUTPUT(resultProb, ALL, NAMED('Probabilities'));
```

And I am unable to visualize the probability distribution of variables.

Tuesday 06/21

I have continued to analyze the dependency of variables in the 'housing' dataset. For the dependency testing between *price* and *sqfeet*, the execution took 3hr35mins. Other variables like *Beds* and *Baths* are found to be highly dependent on each other with a confidence of *0.99*. *Bath* and *Beds* are dependent on *sqfeet* as well with the confidence of 0.63 and 0.59 respectively.

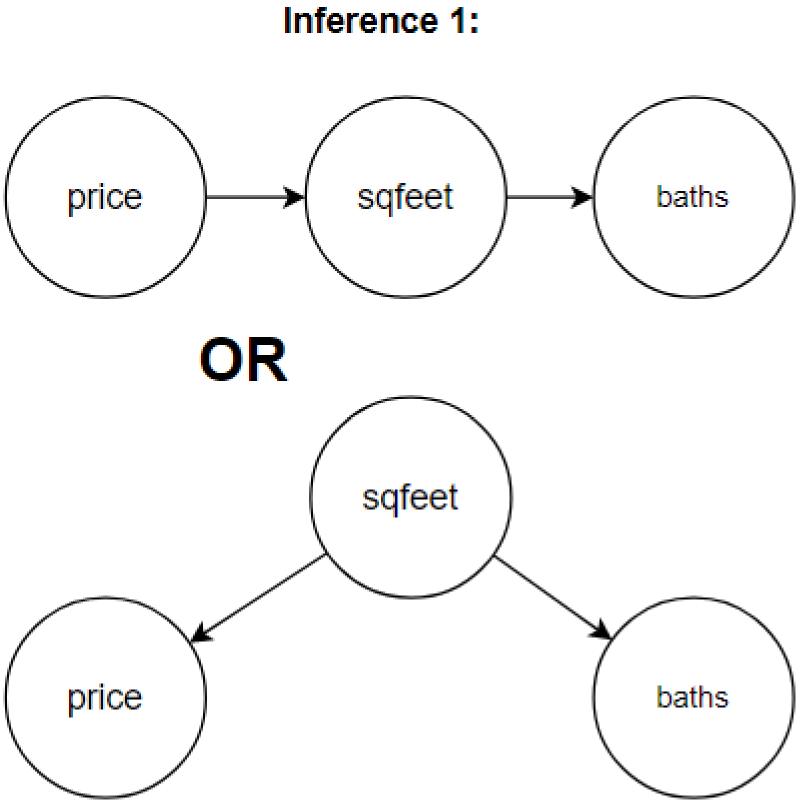
After that, I tried Independence testing for each variable respectively which took about 4 hrs. to complete. And result says that all are dependent on each other.

Wednesday 06/22

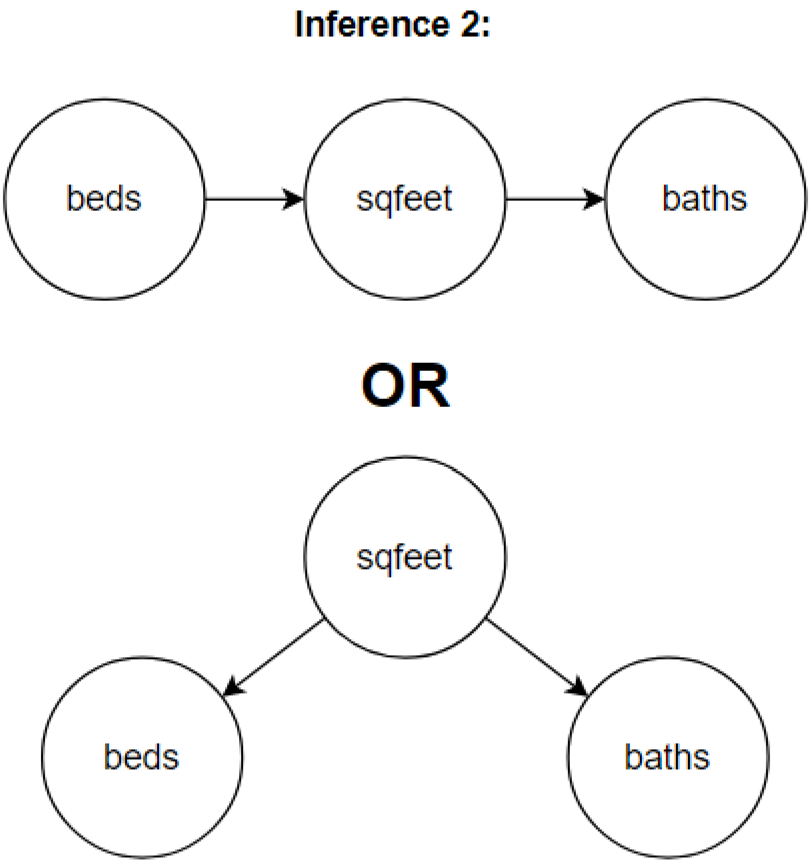
I have asked Roger why the dependence test and independence test is taking longer time. Then we decided to debug the issue in python. We resolved the issue by making some changes in the *because* module.

After that, I have done conditional independence testing, where I have chosen one variable for conditioning. Looking at the results I have made 2 inferences.

Since *price* and *baths* are conditionally independent of given sqfeet, it has to follow either *chain* model or *inverted V* model as shown.



Similarly *beds* and *baths* are conditioned on *sqfeet*.



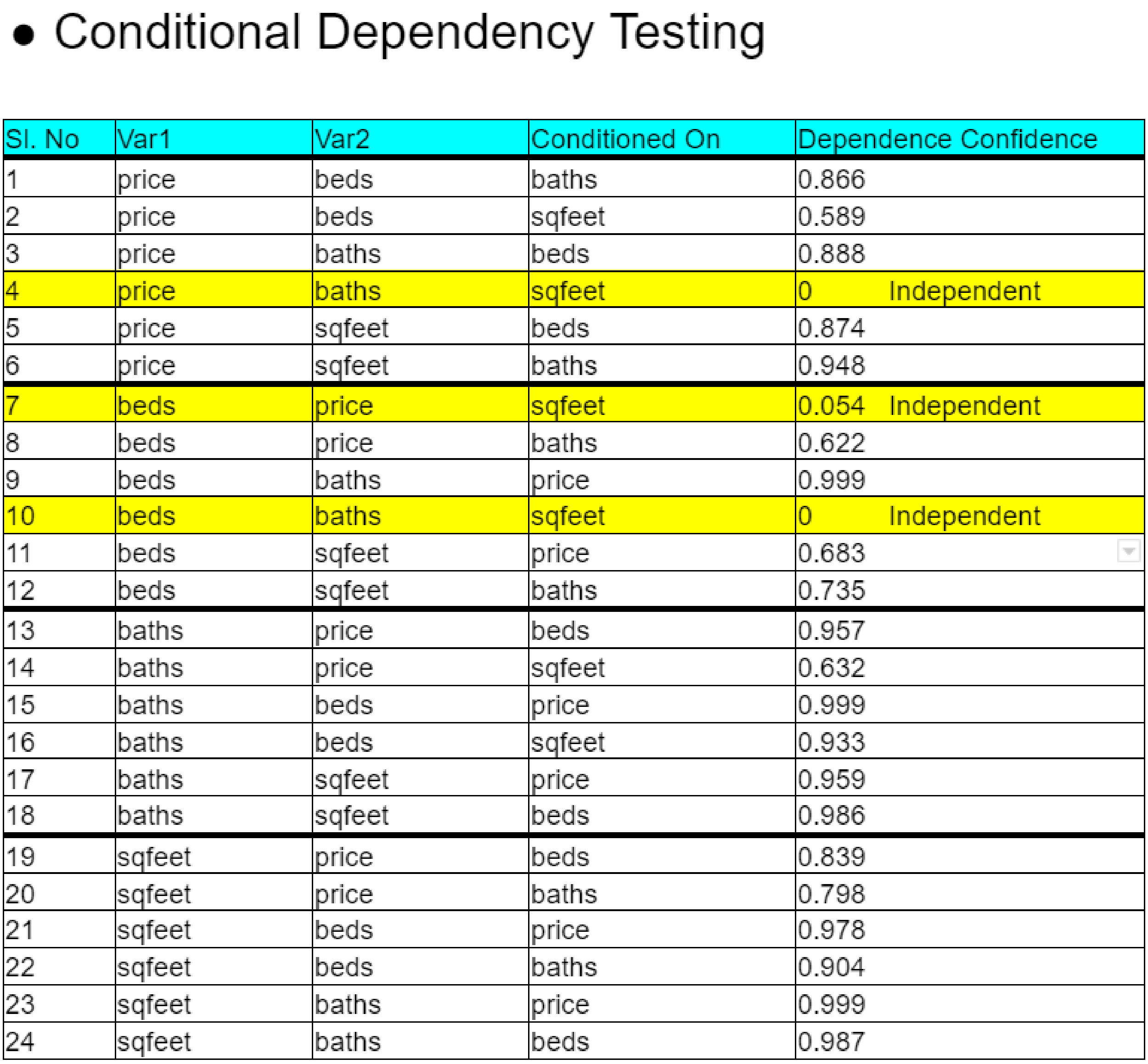
A conflict in the conditional independence results has been found.

1. *beds* and *price* are independent given *sqfeet*. [confidence is 0.054]
2. *price* and *beds* are dependent given *sqfeet*. [confidence is 0.589]

In this case how to conclude from the results?

Thursday 06/23

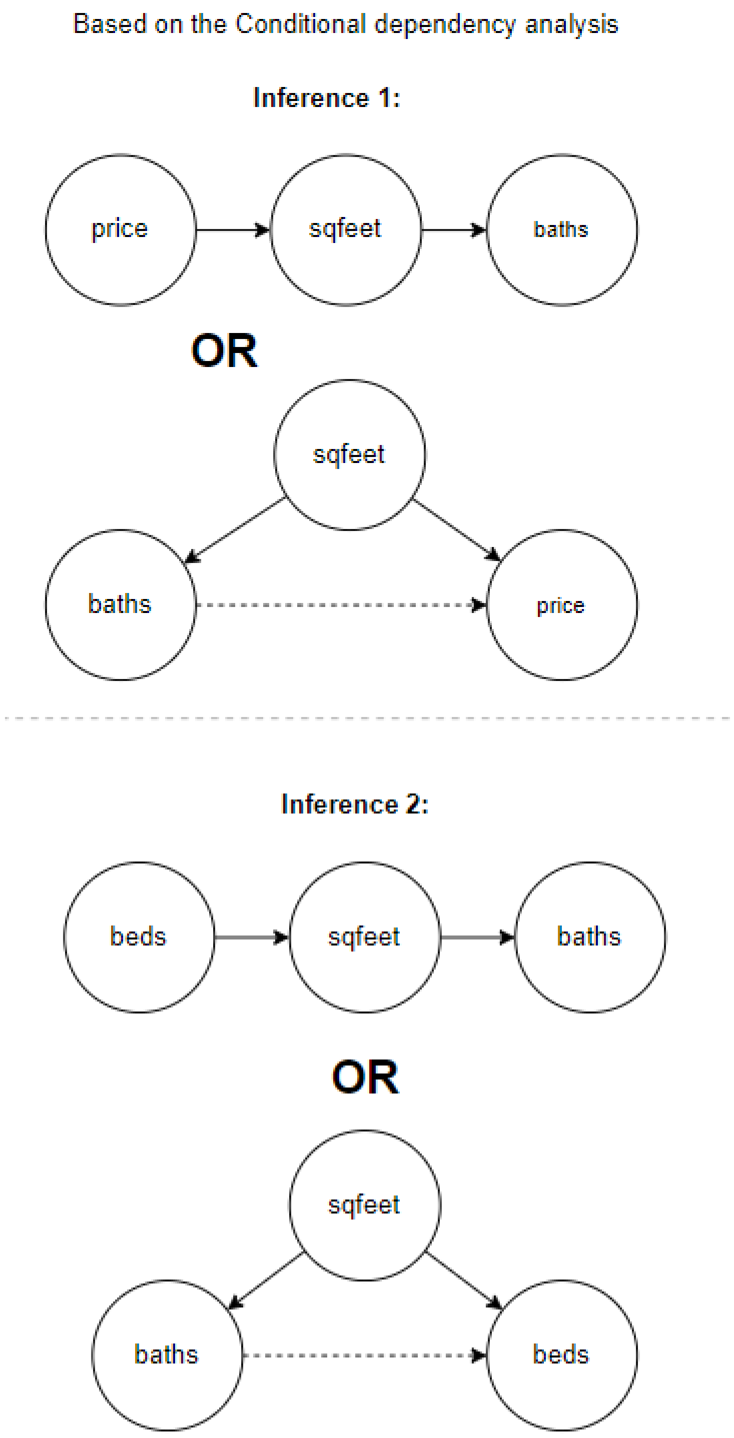
I have tested the dataset for conditional independence and found these results. Using 'prob' test methods, I found that there is no symmetry in the results.



- Conditional Dependency Testing

Sl. No	Var1	Var2	Conditioned On	Dependence Confidence
1	price	beds	baths	0.866
2	price	beds	sqfeet	0.589
3	price	baths	beds	0.888
4	price	baths	sqfeet	0 Independent
5	price	sqfeet	beds	0.874
6	price	sqfeet	baths	0.948
7	beds	price	sqfeet	0.054 Independent
8	beds	price	baths	0.622
9	beds	baths	price	0.999
10	beds	baths	sqfeet	0 Independent
11	beds	sqfeet	price	0.683
12	beds	sqfeet	baths	0.735
13	baths	price	beds	0.957
14	baths	price	sqfeet	0.632
15	baths	beds	price	0.999
16	baths	beds	sqfeet	0.933
17	baths	sqfeet	price	0.959
18	baths	sqfeet	beds	0.986
19	sqfeet	price	beds	0.839
20	sqfeet	price	baths	0.798
21	sqfeet	beds	price	0.978
22	sqfeet	beds	baths	0.904
23	sqfeet	baths	price	0.999
24	sqfeet	baths	beds	0.987

This was the updated inference from the above table.

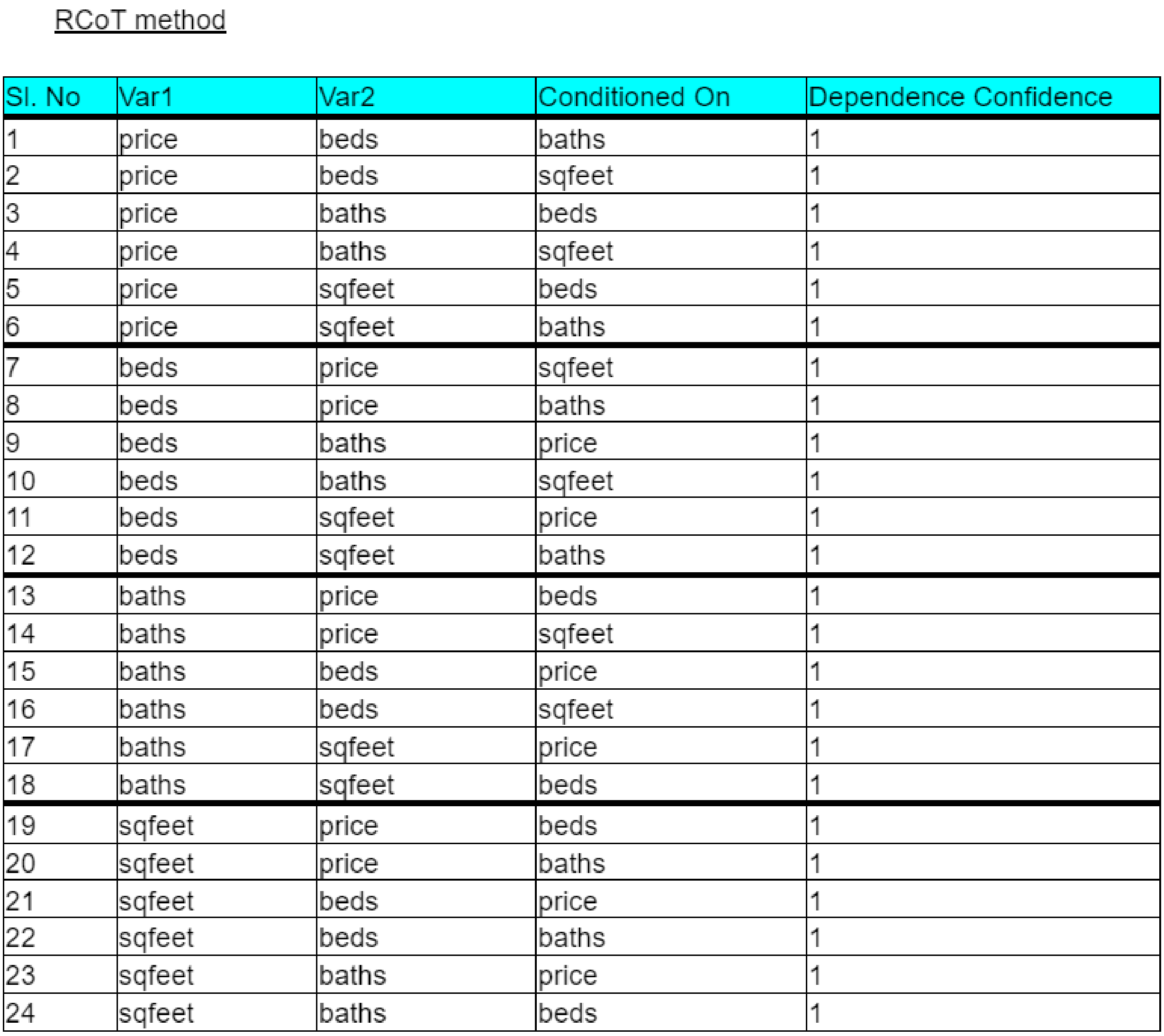


Conditionally (A, B | C) should be the same as (B, A | C), this was violated in the conditional dependency tests. So In the team meeting, it was discussed, and later we pair programmed to add the 'rcot' testing method to the HPCC_Causality bundle.

I have raised a Pull Request to the HPCC_Causality bundle for including the *rcot* method for testing conditionality dependencies.

Friday 06/24

I have tested conditionality dependencies using *rcot* method and found the following results.

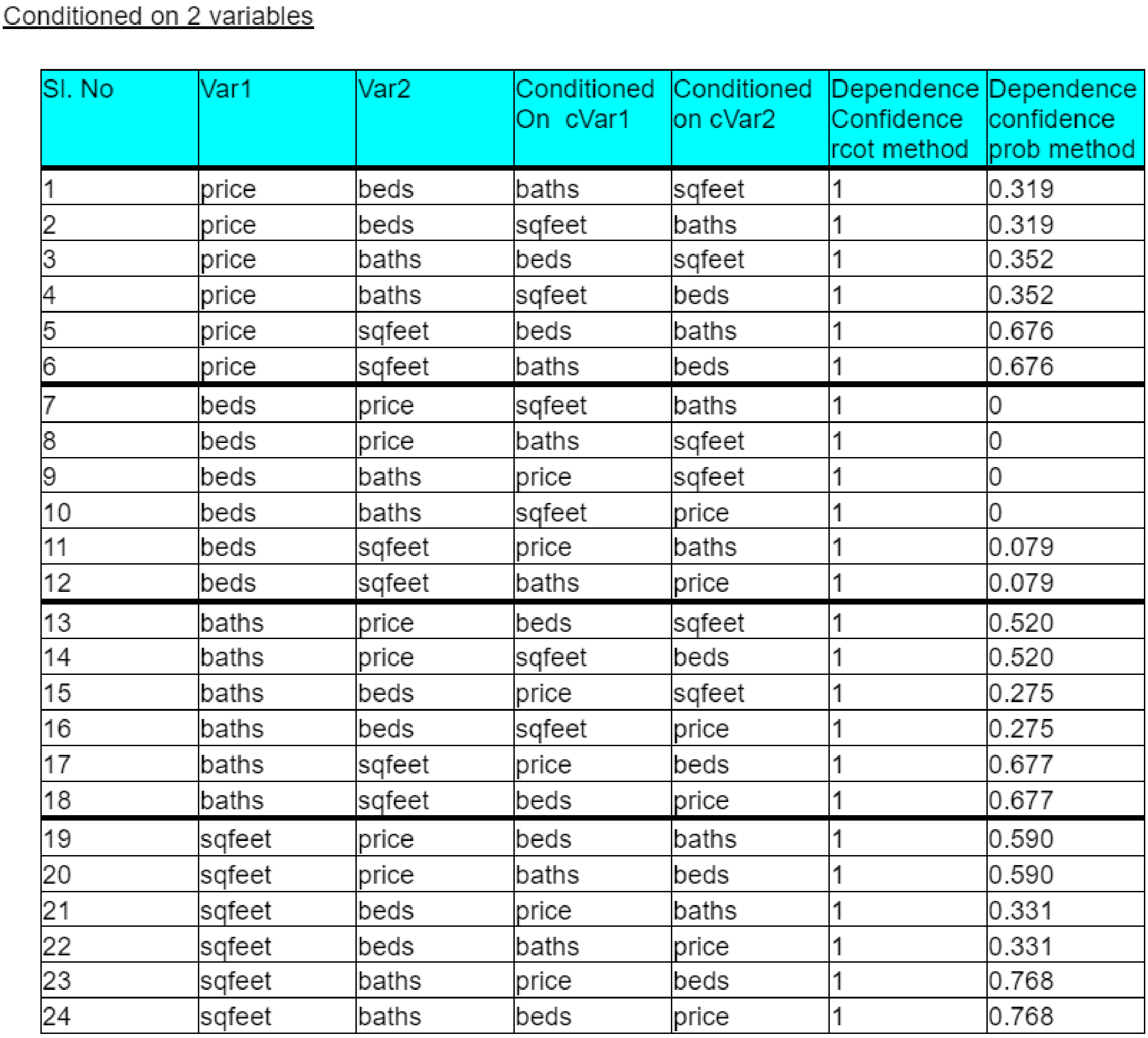


RCoT method

Sl. No	Var1	Var2	Conditioned On	Dependence Confidence
1	price	beds	baths	1
2	price	beds	sqfeet	1
3	price	baths	beds	1
4	price	baths	sqfeet	1
5	price	sqfeet	beds	1
6	price	sqfeet	baths	1
7	beds	price	sqfeet	1
8	beds	price	baths	1
9	beds	baths	price	1
10	beds	baths	sqfeet	1
11	beds	sqfeet	price	1
12	beds	sqfeet	baths	1
13	baths	price	beds	1
14	baths	price	sqfeet	1
15	baths	beds	price	1
16	baths	beds	sqfeet	1
17	baths	sqfeet	price	1
18	baths	sqfeet	beds	1
19	sqfeet	price	beds	1
20	sqfeet	price	baths	1
21	sqfeet	beds	price	1
22	sqfeet	beds	baths	1
23	sqfeet	baths	price	1
24	sqfeet	baths	beds	1

I have tested conditionality independencies for variables of housing dataset conditioned on 2 variables using both *prob* and *rcot* methods. I have found some interesting results like both methods giving different results for a few queries.

Using *rcot* method I have found that all are dependent but using *prob* method few are not.



Conditioned on 2 variables

Sl. No	Var1	Var2	Conditioned On cVar1	Conditioned on cVar2	Dependence Confidence rcot method	Dependence confidence prob method
1	price	beds	baths	sqfeet	1	0.319
2	price	beds	sqfeet	baths	1	0.319
3	price	baths	beds	sqfeet	1	0.352
4	price	baths	sqfeet	beds	1	0.352
5	price	sqfeet	beds	baths	1	0.676
6	price	sqfeet	baths	beds	1	0.676
7	beds	price	sqfeet	baths	1	0
8	beds	price	baths	sqfeet	1	0
9	beds	baths	price	sqfeet	1	0
10	beds	baths	sqfeet	price	1	0
11	beds	sqfeet	price	baths	1	0.079
12	beds	sqfeet	baths	price	1	0.079
13	baths	price	beds	sqfeet	1	0.520
14	baths	price	sqfeet	beds	1	0.520
15	baths	beds	price	sqfeet	1	0.275
16	baths	beds	sqfeet	price	1	0.275
17	baths	sqfeet	price	beds	1	0.677
18	baths	sqfeet	beds	price	1	0.677
19	sqfeet	price	beds	baths	1	0.590
20	sqfeet	price	baths	beds	1	0.590
21	sqfeet	beds	price	baths	1	0.331
22	sqfeet	beds	baths	price	1	0.331
23	sqfeet	baths	price	beds	1	0.768
24	sqfeet	baths	beds	price	1	0.768

But no proper conclusions can be drawn.

HOME | Timeline | Previous Week | Next Week