Test Result on Property

1. Result on Sequence

1.1 delete activity from sequence

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 1.1 | model\_01\_sequence.pnml | 01\_log\_sequence\_02.xes | result-02\_log\_parallelism\_08.xlsx | result-neg-weight-02\_log\_sequence\_02 .png | Recall keep 1.0  Precision and accuracy goes up,  specificity goes up  TP same, FP down, TN up, FN 0 |
| result-ext-weight-02\_log\_sequence\_02 .png | Recall 1.0, others goes down, accuracy and precision overlap  TN same, Fp up, TN down, FN 0. |
| result-pos-weight-02\_log\_sequence\_02 .png | All keep stable, recall 1.0, precision and accuracy overlap  All kept the same values..  But the reason is that, if the factors from existing and negative balances each other, then, existing keeps, or new model generates. So the pos |

2. Result on Parallel

2.1 Add activities in parallel relation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 2.1 | model\_02\_parallism.pnml | 02\_log\_parallelism\_08.xes | result-02\_log\_parallelism\_08.xlsx | result-neg-weight-02\_log\_parallelism\_08.png | Recall , accuracy down,  F, Precision, Specificy overlapped stay stable |
| result-ext-weight-02\_log\_parallelism\_08.png | Recall varies a lot, overlapped with others, while it ranges from 0.5–0.9  Precision, F has missing values, but other part stays the 1.0  Specificity stays always at 1.0 |
| result-pos-weight-02\_log\_parallelism\_08.png | Recall, accuracy vary a lot.  Precision, F has missing values, but other part stays the 1.0  Specificity stays always at 1.0 |
|  |  |  |  |  |  |

The effect at existing model varies so much, is because of the model generated by weighted data. It depends on the deviations of model and actual data. If they deviate a lot, then no need to keep the existing model;; Else, there is some ideas to keep the model.

To prove the significance of the existing model, we need to test the similarity of those models. We can compare it with the inductive miner and check their result.

# Analysis:

the figures show that with the weights of negative increases, the model becomes more power to address the accuracy. But the model is generated due to balance of existing model, positive and negative instances.

But we can see that, positive and

2.2 change parallel to xor

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 2.2 | model\_02\_parallism.pnml | 02\_log\_parallelism\_02.xes | result-02\_log\_parallelism\_02.xlsx  Here we delete the effect of mixed standard, and now, we can have the right groups here. | result-neg-weight-02\_log\_parallelism\_02.png | ~~Recall , accuracy down,~~  ~~F, Precision, Specificy overlapped stay stable~~  With the negative weights goes, it grows up all the measurements.  Reason: negetive weights more, then  Precision and accuracy: the true positive got support and blocked from the model.  Tp up, FP down, TN up, FN down. |
| result-ext-weight-02\_log\_parallelism\_02.png | ~~Precision, and F are with NaN data, but the total frequency goes down.. while Fscore stays the same.~~  ~~Accuracy shows the similar changes, go down.~~  ~~Recall vabirate a lot but tendency goes down.~~  ~~Specificity has the same tendency like recall, but higher value.~~  With the existing values go up, the values go down, the reason for this??? Recall goes down is because the true positive goes down. Why?? The model needs change, it doesn’t fit the model anymore. Then TP smaller.  Accuracy goes down and precision: TP less and FP increases, but why?? Existing model deviates from the old model. True negative goes down. Because the model supports more negative instances.  Specificity: TN goes down and the models is unable to reject the resuls. |
| result-pos-weight-02\_log\_parallelism\_02.png | ~~Recall varies a lot, unstable,accuracy higer than precision, but varied also a lot.~~  ~~Specificity vabriate above 0.60. precision with missing values, but show a slow going up. Vabirate .~~  ~~F value varies a lot and stays around 0.0 value.~~  Positive goes up, the other goes up, except the specificity stays same. |

2.3 change parallel to sequence with overlapped labels.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 2.3 | model\_02\_parallism.pnml | 02\_log\_parallelism\_06.xes | result-02\_log\_parallelism\_06.xlsx | result-neg-weight-02\_log\_parallelism\_06.png | Accuracy and precision overlaps, the other goes up, recall and specificity.  TP down, Fp down, TN up, FN up. |
| result-ext-weight-02\_log\_parallelism\_06.png | Recall goes a bit up slowly, while accuracy and precision overlap and go down. The same wityh specificity.  TP up, FP up, TN down, FN down |
| result-pos-weight-02\_log\_parallelism\_06.png | Recall goes a bit up slowly, while accuracy and precision overlap and go down. The same wityh specificity.  TP up, FP up, TN down, FN down |

3. Result on Loop

3.1 delete loop

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 3.1 | model\_03\_loop\_03.pnml | 03\_log\_loop\_07.xes | result-03\_log\_loop\_07.xlsx | result-neg-weight-03\_log\_loop\_07.png | Accuracy and precision overlaps and go up. Recall keep stable.  TP same, Fp down, TN up, FN 0.  Why false negative 0?? The model not change any positive choices, so it adjust the effect to avoid the negative isntances. Because TN up. |
| result-ext-weight-03\_log\_loop\_07.png | Recall keeps the same, while accuracy and precision overlap and go down. The same down with specificity.  TP same, FP up, TN down, FN same  Ext keeps the model and damages the precision and accuracy. |
| result-pos-weight-03\_log\_loop\_07.png | All keep 1.0.  Because the TN and TN keeps same, other 0. |

3.2

4. Result on Xor

4.1 delete xor branch from model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 4.1 | model\_05\_xor\_02.pnml | 05\_log\_xor\_03.xes | ?? Not recorded.. later to remember to record | result-neg-weight-05\_log\_xor\_03.png | Accuracy and precision overlaps and go up. Recall keep stable with 1.0.  TP same, Fp down, TN up, FN 0.  Why false negative 0?? The model not change any positive choices, so it adjust the effect to avoid the negative isntances. Because TN up. |
| result-ext-weight-05\_log\_xor\_03.png | Recall keeps the same with 1.0, while accuracy and precision overlap and go down. The same down with specificity but lower than precision.  TP same, FP up, TN down, FN same with 0  Ext keeps the model and damages the precision and accuracy. |
| result-pos-weight-05\_log\_xor\_03.png | All keep 1.0.  Because the TN and TN keeps same at most number, other 0. |

4.2 add long-term dependency of the model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 4.2 | model\_04\_gdep\_01.pnml | 04\_log\_gdep\_01.xes | result-model-04\_log\_gdep\_01.png with long-term dependency  result-model-04\_log\_gdep\_01\_no\_events.png  With ext: pos: neg = 0.05, 0.94,1.0  , not events are allowed  0:0💯100  with 0.7:0.123:0.1, the existing model hols the biggest percentage, so it keeps original.  The other is due to one factor over others, so it deletes or disapper.  We need to keep in mind, if the neg is over all others, then the effect loses, if the neg includes all the factors in pos, ext.. Else no | result-neg-weight-04\_log\_gdep\_01.png | Recall goes down with steps.  Precision goes up , ending with 1.0  Accuracy has a tendency up, but with some jumps  Specificity: goes up, but lower than precision.  TP down, Fp down, TN up, FN up. |
| result-ext-weight-04\_log\_gdep\_01.png | Recall keeps growing.  Precision overlapes with accuracy, firstly up and then down  Specifity up and down..  TP up, FP firstly down and then up, TN up and down, FN down |
| result-pos-weight-04\_log\_gdep\_01.png | Recall up, precision and accuracy up , but specifity fitst down then up, all have tendency up with positive  TN up, TP down, TN up, FN down |

4.3 delete long-term dependency on it

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 4.1 | model\_05\_xor\_02.pnml | 05\_log\_xor\_02.xes | result-model-04\_log\_gdep\_02.png  It always generate a model without long-term dependency |  |  |
|  |  |
|  |  |

4.2 change the structure of them into others..

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Model name | Log name | Result data | Figure type and name | Figure Description |
| 4.1 | model\_05\_xor\_02.pnml | 05\_log\_xor\_06.xes | result-model-04\_log\_gdep\_06.png  o.23:0.4:0.7,silent model  Model without lt is the existing model with pos fill the conditions, no model should be there.  But it can not keep the model like original, because it deletes it while we deal with those model..  How to keep the long-term dependency on the original model?? I need to seek a way to do it and the compare with the effect with other models. Decide at last!!  How to recognize the long-term dependency in the model?? We really need to consider them!!! | result-neg-weight-05\_log\_xor\_06.png | Recall goes down with steps.  Precision goes up , ending with 1.0  Accuracy has a tendency up, but with some jumps  Specificity: goes up, but lower than precision.  TP down, Fp down, TN up, FN up. |
| result-ext-weight-05\_log\_xor\_06.png | Recall keeps growing.  Precision overlapes with accuracy, firstly up and then down with steps  Specifity up and down..  TP up, FP firstly down and then up, TN up and down, FN down |
| result-pos-weight-05\_log\_xor\_06.png | Recall up, precision and accuracy up , but specifity fitst down then up, all have tendency up with positive  TN up, TP down, TN up, FN down |

Test on synthetic data

here we check the effect of our methods on the simulated data, but how to do this?? It differs from the property test where we don’t need to have the comparison with other methods. But test on the whole, should we compare with others?? Not really??

Or we should test on the difference between them, with Inductive Miner, Repair Model and Dee’s method, the situations where others can not handle!!

Synthetic data and real life should be considered as one type. If we only get the result, it’s fine. But should we test on what ?? I can ask the opinions of Bas.