CS5340 Lab 1: Belief Propagation and Maximal Probability


```
Name: Niharika Shrivastava
Email: niharika.shrivastava@u.nus.edu
Student ID: A0254355A
```


1. `factor_product()` and `factor_sum()`

Result is the element-wise multiplication (or addition in case of `factor_sum()`) of values at `A.val[idxA]` and `B.val[idxB]`.

2. `factor_marginalize()` and `factor_max_marginalize()`

Marginalize every variable from the factor one-by-one. Remove the variable and its cardinality from the original factor to get the final factor variable and cardinality.

Sum (or take maximum in case of `factor_max_marginalize()`) values of all the rows in the original factor which equals one row in the final factor (without the marginalized variable value). E.g., `X0` to be marginalized:

```
Original factor:             Marg assignment:    Final factor(sum): Final factor(max):
X0  X1  P(X0, X1)            X1  P(X0, X1)       X1  P(X1)          X1  P(X1)
0   0   0.2                  0   0.2             0   0.5            0   0.3 (argmax={X2=0, X0=1})
1   0   0.3 (argmax={X2=0})  0   0.3             1   0.9            1   0.5 (argmax={X0=0})
0   1   0.5                  1   0.5
1   1   0.4 (argmax=None)    1   0.4
```

3. `observe_evidence()`

Loop through every factor and select the ones where the evidence variable is present. Set probablity of all its assignments to `0` if it doesn't contain the evidence.

4. `compute_joint_distribution()`

Do a `factor_product` of all factors in the list sequentially.

5. `compute_marginals_naive()`

Observe evidence for all factors. Compute its joint probability. Marginalize over all the variables minus the given variable. Normalize the probabilities.

6. `compute_marginals_bp()` Do a DFS on the graph and store messages such that:

```
messages[a][b] ==> message from node a to node b, followed by inference on node b
```

Collect all messages from leaf to root nodes. Distribute all messages from root to the leaf nodes. Do an inference for all variables in `V`.

7. `map_eliminate()`

- Observe evidence for all factors. Convert all proabilities into its log values.
- Generate graph from the factors. Select `root=Node(0)`.
- Collect all messages from leaf to root nodes. Perform inference on the root node. These steps are enough to give us the:
    - Max probability of the entire graph as maximum probability remains the same no matter what the query node is.
    - Max probability configuration as `val_argmax` is populated while computing the marginals every time.
- Remove the evidence from the max decoding configuration and add for the root node.