Approximate Inference

Phu Sakulwongtana

1 Graphical Model

1.1 Introduction

Definition 1.1. (Types of Graph) There are several kind of graphs that we can use to model the probability distribution: factor graph, undirected graph, and directed graph. Node corresponds to the random variables and the edge in graph indicates statisical dependence between varible.

Definition 1.2. (Dependencies) For the random variable X, Y, V, where we have:

- *Conditional Independence*: $X \perp\!\!\!\perp Y|V$ iff $P(X|Y,V) = P(X|V)$ provided that $P(Y,V) > 0$. We can see that furthermore that:

  $$P(X,Y|V) = P(X|Y,V)P(Y|V) = P(X|V)P(Y|V)$$

  Please note that, this can generalize the symbol to the sets of random variables as:

  $$\mathcal{X} \perp\!\!\!\perp \mathcal{Y}|\mathcal{V} = \{X \perp\!\!\!\perp Y|\mathcal{V} : \forall X \in \mathcal{X}, \forall Y \in \mathcal{Y}\}$$

- *Marginal Independence*: $X \perp\!\!\!\perp Y$ is equivalent to $X \perp\!\!\!\perp Y|\emptyset$ and $P(X,Y) = P(X)P(Y)$

Definition 1.3. (Factor Graph) Factor Graph is a directed graphical representation of the factorized model structure, where each square indicates the factor over the linked variables:

$$P(\mathcal{X}) = \frac{1}{Z}\prod_j f_j(\mathcal{X}_{C_j})$$

where we have the following components:

- $\mathcal{X} = \{X_1, \ldots, X_k\}$
- $\mathcal{X}_S = \{X_i : i \in S\}$
- j is index that indicates the factor C_j that contains all indicies of variable adjecent to factor j
- f_j is factor function
- Z is normalization constant

The conditional independent is defined by $X \perp\!\!\!\perp Y|\mathcal{V}$ if every path between X and Y contains some $V \in \mathcal{V}$ (this can be shown that).

Remark 1. **(Conditional Distribution)** Now, if every path between X and Y contains some $V \in \mathcal{V}$, then there exists a factorization. We have the following joint distribution

$$P(X,Y,\mathcal{V},\ldots) = \frac{1}{Z}g_X(X,\mathcal{V}_X,\mathcal{Q}_X)g_Y(Y,\mathcal{V}_Y,\mathcal{Q}_Y)g_R(\mathcal{Q}_R,\mathcal{V}_R)$$

where $\mathcal{V}_X, \mathcal{V}_Y, \mathcal{V}_R \subseteq \mathcal{V}$ and the set containing $\mathcal{Q}_X, \mathcal{Q}_Y, \mathcal{Q}_R$ are disjoint. The conditonal is:

$$P(X|Y,\mathcal{V},\dots) = \frac{P(X,Y,\mathcal{V},\dots)}{P(Y,\mathcal{V},\dots)} = \frac{\frac{1}{Z}g_X(X,\mathcal{V}_X,\mathcal{Q}_X)g_Y(Y,\mathcal{V}_Y,\mathcal{Q}_Y)g_R(\mathcal{Q}_R,\mathcal{V}_R)}{\sum_{X'}\frac{1}{Z}g_X(X,\mathcal{V}_X,\mathcal{Q}_X)g_Y(Y,\mathcal{V}_Y,\mathcal{Q}_Y)g_R(\mathcal{Q}_R,\mathcal{V}_R)}$$
$$= \frac{g_X(X,\mathcal{V}_X,\mathcal{Q}_X)}{\sum_{X'}g_X(X',\mathcal{V}_X,\mathcal{Q}_X)}$$

One the RHS doesn't depend on Y as it follows that $X \perp\!\!\!\perp Y|\mathcal{V}$.

Definition 1.4. (Markov Blanket) $\mathcal{V}$ is markov blanket for X iff $X \perp\!\!\!\perp Y|\mathcal{V}$ for all $Y \notin \{X \cup \mathcal{V}\}$

Remark 2. Each variable X is conditionally independent of all non-neighbourhood given its neighbourhood as we have:

$$X \perp\!\!\!\perp Y|\,\mathrm{ne}(X) \qquad \forall Y \notin \{X \cup \mathrm{ne}(X)\}$$

All neighbourhood $\mathrm{ne}(X)$ is markov blanket of X. Please note that it is minimal of such set (markov blanket), which is called *markov boundary*.

Definition 1.5. (Cliques) Cliques is fully connected subgraph, whiel the maximal clique is a clique that isn't contains in the other cliques.

Definition 1.6. (Undirected Graphical Model) The undirected graphical model is a direct representation of conditional independent and nodes are connected iff they are conditonally dependent given all others. The joint probability factors over maximal clique C_j of the graph is given by:

$$P(\mathcal{X}) = \frac{1}{Z}\prod_j f_j(\mathcal{X}_{C_j})$$

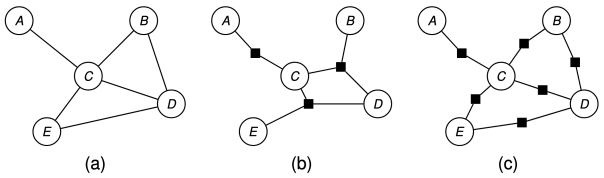
We have the following dependencies properties:

- $X \perp\!\!\!\perp Y|\mathcal{V}$ if every path between X and Y contains some node $V \in \mathcal{V}$
- Each variable X is conditionally independent of all non-neighbour node given its neighbourhood nodes:

  $$X \perp\!\!\!\perp Y|\mathrm{ne}(X) \qquad Y \in \{X \cup \mathrm{ne}(X)\}$$

  And so, the neighbours is a markov blanket.

Remark 3. **(Factor Graph vs Undirected Model)** Consider 3 difference types of graph, we can see that each nodes has same neighbour:

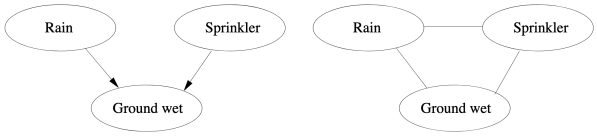


Each graph represents exactly the same conditional independent relationship. However, the maximal factorization differs, suppose we have for each variable K possible values:

- (a) can't distiguish between these (we will adopt the (b) to be safe)
- (b) has 2 three-way factor. This is represented in $\mathcal{O}(K^3)$-size table.
- (c) has only pairwise factors. This is represented in $\mathcal{O}(K^2)$-size table.

This means that the factor graphs have richer expressive power than undirected graphical models. But the factors can't be determined by testing for conditional independent.

Remark 4. **(Limitation of Undirected Graphical Model)** Undirected and Factor graph fails to capture the dependencies as the pair of variables that may be connected because they are some other variable that depends on them, for example:



If the ground is damped, it may suggest that it was rain, but if we see a sprinkler, then this explain away the damp, thus reduce the our belief of the rain into the prior. For example:

$$R \perp\!\!\!\perp S|\emptyset \quad \text{but} \quad R \perp\!\!\!\perp S|G$$

This is where there is difference between marginal and conditional independent.

Definition 1.7. (DAG Graphical Model) A directed acyclic graphical model (DAG) represents a factorization of the joint probability distribution in terms of condtional:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|X_{\text{pa}(i)})$$

where $\text{pa}(\cdot)$ is the parent of node i. DAG models are called Baysian network.

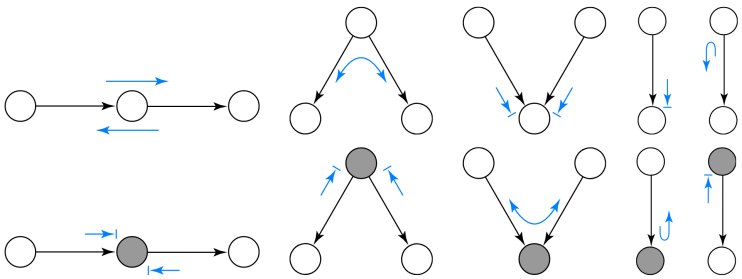
Proposition 1.1. *The conditional Independence between the graph is more complicated than the undirected graph i.e $X \perp\!\!\!\perp Y|\mathcal{V}$. If we consider every undirected path between X and Y, the path is blocked by $\mathcal{V}$ if there is a node V on the path such that:*

- *V has convergnece arrows $\rightarrow V \leftarrow$ on the path and neighbour V nor its descendent are in $\mathcal{V}$*
- *V doesn't have convergnece arrow $\leftarrow V \rightarrow$ or $\rightarrow V \rightarrow$ and $V \in \mathcal{V}$*

If all paths are blocked, then $\mathcal{V}$ is D-separated between X and Y, and so $X \perp\!\!\!\perp Y|\mathcal{V}$. Furthermore, the markov boundary to be:

$$\{\text{pa}(X) \cup \text{ch}(X) \cup \text{pa}(\text{ch}(X))\}$$

Proof. We can see that the conditional independence of the directed graphical model (for example $A \perp\!\!\!\perp B|\mathcal{D}$) can be modeled as the passing of "ball", in which 2 variables (A, B) aren't independence if there is a way that a ball can be passed between them. We will mark the nodes in $\mathcal{V}$ as shaded. There are 10 simple rules:



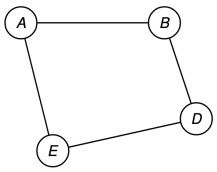
Most of the rules are straightforward to see why it is enforced.

- The first column: Both variables are separated by a middle node, then both of the are independent of each other, thus unable to pass the "ball" to each other, meaning that they are independence given the middle node. (This represents the divergence arrow rule)

- Similar explaination can be done in the second column of the image. (This also represents the divergence arrow rule)
- For the third column (explaining away), we can see that both of the nodes are independence given nothing, however, they becomes dependence once the middle node is shown. This is a reflection of the convergnece arrow rule.
- Finally, the last 2 columns are boundary rule, which is also straightforward to see why it is enforced.

Thus, we have the reason why the rules above are used. □

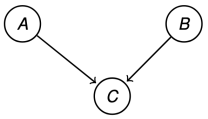
Remark 5. **(Differences Between DAG and Factor Graph)** There are some types of graphs that DAG can represent its probability distribution, which is:



This is the only graph that that DAG can't represent. This is because there will always be 2 non-adjecent parent sharing the same child, which implies that:

- The variables are dependence in DAG
- But independence in undirected graph.

On the other hand, no undirected or factor graph can represent the following DAG and only these:



This follows from the previous analysis on the explaining away and marginal independence.

Definition 1.8. (Family of Distribution) Each graph $\mathcal{G}$ implies a set of conditional independence statement: $\mathcal{C}(G) = \{X_i \perp\!\!\!\perp Y_i | \mathcal{Y}\}$. Each set $\mathcal{C}$ defines a family of distribution that satisfies all statement in $\mathcal{C}$:

$$P_{\mathcal{C}(G)} = \{P(\mathcal{X}) : P(X_i, Y_i|\mathcal{V}) = P(X_i|\mathcal{V}_i)P(Y_i|\mathcal{Y}_i) \text{ for all } X_i \perp\!\!\!\perp Y_i|\mathcal{V}_i \text{ in } \mathcal{C}\}$$

Similarly, we have family distribution in the functional form i.e:

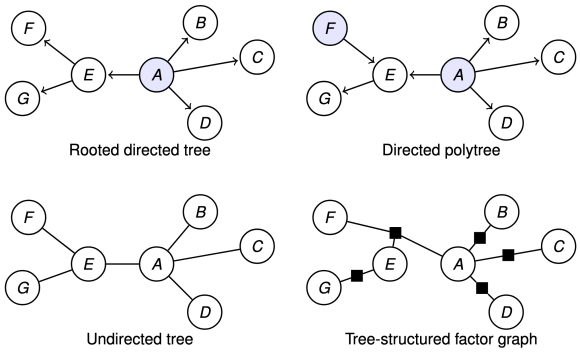
$$P_G = \left\{ P(\mathcal{X}) : \frac{1}{Z}\prod_j f_j(\mathcal{X}_{C_j}) \text{ for some non-negative function } f_j \right\}$$

Remark 6. **(Family of Distributions)** We can consider the following facts:

- For directed graph: $P_G = P_{\mathcal{C}(G)}$
- For undirected graph: $P_G = P_{\mathcal{C}(G)}$ if all distribution are positive i.e $P(\mathcal{X}) > 0$ for all variable of $\mathcal{X}$
- Factor graphs are more expressive as for every undirected graph G_1, there is a factor graph G_2 such that: $P_{G_1} = P_{G_2}$ but not in other direction.

Adding edge implies removing conditional independency statement and thus enlarging family of distribution.

Remark 7. For the next few propositions, we will consider difference kinds of graphical models, which can be shown to interchange with each others:



Proposition 1.2. ***(Polytree → Tree-Structured Factor Graph)*** *For DAG that has more than one root, we consider:*

$$P(\mathcal{X}) = \prod_i P(X_i|X_{pa(i)}) = \prod_i f_i(X_{C_i})$$

where $C_i = i \cup \text{pa}(i)$ *and* $f(X_{C_i}) = P(X_i|X_{\text{pa}(i)})$

Proposition 1.3. ***(Undirected Tree → Factor Graph)*** *Since all undirected tree have maximal clique of size* 2*, it is equivalent to factor graph with pairwise factor:*

$$P(\mathcal{X}) = \frac{1}{Z} \prod_{\text{edge}(i,j)} f_{(i,j)}(X_i, X_j)$$

Proposition 1.4. ***(Rooted Directed Tree → Undirected Tree)*** *The distribution for single rooted directed tree can be written as a product of pairwise factor of the undirected tree:*

$$P(\mathcal{X}) = P(X_r) \prod_{i \neq r} P(X_i|X_{\text{pa}(i)}) = \prod_{\text{edge}(i,j)} f_{(i,j)}(X_i, X_j)$$

Proposition 1.5. ***(Undirected Tree → Rooted Directed Tree)*** *Choose arbitary node* X_r *and set it as a root, which we will point every arrow away from it. Compute the conditional in the DAG as:*

$$P(\mathcal{X}) = P(X_r) \prod_{i \neq r} P(X_i|X_{\text{pa}(i)}) = P(X_r) \prod_{i \neq r} \frac{P(X_i, X_{\text{pa}(i)})}{P(X_{\text{pa}(i)})} = \frac{\prod_{\text{edge}(ij)} P(X_i, X_j)}{\prod_{\text{nodes}(i)} P(X_i)^{\deg(i)-1}}$$

2 Belief Propagation

Remark 8. We want to calculate the marginal distribution of a single node $P(X_i)$ and on the edge $P(X_i, X_j)$ based on the undirected graphical model, which we would need to use a belief propagation.

Proposition 2.1. ***(Marginal Distribution)*** *The marginal distribution can be calculated locally as:*

$$P(X_i) = \prod_{X_j \in \text{ne}(i)} M_{j \to i} \quad \textit{where} \quad M_{j \to i} = \sum_{X_j} f_{ij}(X_i, X_j) \prod_{X_k \in \text{ne}(X_j) \setminus \{X_i\}} M_{k \to j}(X_j)$$

Proof. For each neighbourhood X_j of X_i define a disjoint subtree $T_{j\to i}$ that split by X_i:

$$\begin{aligned}
P(X_i) &= \sum_{\mathcal{X}\setminus\{X_i\}} P(\mathcal{X}) \propto \sum_{\mathcal{X}\setminus\{X_i\}} \prod_{(i,j)\in\mathcal{E}_T} f_{ij}(X_i, X_j) \\
&= \sum_{\mathcal{X}\setminus\{X_i\}} \prod_{X_j\in\text{ne}(X_i)} f_{ij}(X_i, X_j) \prod_{(i',j')\in\mathcal{E}_{T_{j\to i}}} f_{i'j'}(X_{i'}, X_{j'}) \\
&= \prod_{X_j\in\text{ne}(i)} \left(\sum_{\mathcal{X}_{T_{j\to i}}} f_{ij}(X_i, X_j) \prod_{(i',j')\in\mathcal{E}_{T_{j\to i}}} f_{i'j'}(X_{i'}, X_{j'}) \right)
\end{aligned}$$

The last equality comes from the splitting between subtrees, as they are disjoint. $\mathcal{X}_{T_{j\to i}}$ denotes each edge in the subtree. Let's consider the message:

$$\begin{aligned}
M_{j\to i} &= \sum_{\mathcal{X}_{T_{j\to i}}} f_{ij}(X_i, X_j) \prod_{(i',j')\in\mathcal{E}_{T_{j\to i}}} f_{i'j'}(X_{i'}, X_{j'}) \\
&= \sum_{X_j} f_{ij}(X_i, X_j) \sum_{\mathcal{X}_{T_{j\to i}}\setminus\{X_j\}} \prod_{(i',j')\in\mathcal{E}_{T_{j\to i}}} f_{i'j'}(X_{i'}, X_{j'}) \\
&= \sum_{X_j} f_{ij}(X_i, X_j) \prod_{X_k\in\text{ne}(X_j)\setminus X_i} M_{k\to i}(X_j)
\end{aligned}$$

The second equality comes from the fact that the factor of X_i is the root of the tree and for any nodes that connection to the factors. If we consider:

$$\begin{aligned}
\sum_{\mathcal{X}_{T_{j\to i}}\setminus\{X_j\}} \prod_{(i',j')\in\mathcal{E}_{T_{j\to i}}} f_{i'j'}(X_{i'}, X_{j'}) &\propto P_{T_{j\to i}}(X_j) \\
&\propto \prod_{X_k\in\text{ne}(X_j)\setminus X_i} M_{k\to i}(X_j)
\end{aligned}$$

This is due to recursive property of the message passing. □

Proposition 2.2. ***(Pairwise Marginal)***

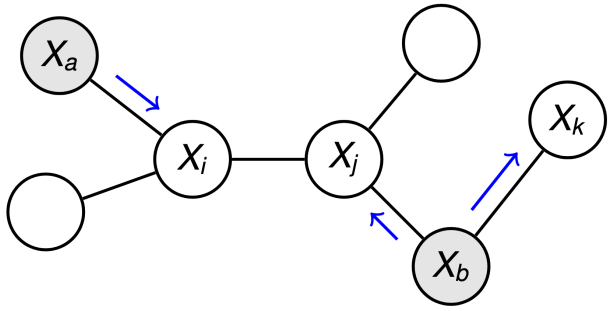
$$P(X_i, X_j) = f_{ij}(X_i, X_j) \prod_{X_k\in\text{ne}(X_j)\setminus\{X_i\}} M_{k\to j}(X_j) \prod_{X_k\in\text{ne}(X_i)\setminus\{X_j\}} M_{k\to i}(X_i)$$

Proof. We consider:

$$\begin{aligned}
P(X_i, X_j) &= \sum_{X\setminus\{X_i,X_j\}} P(\mathcal{X}) \propto \sum_{X\setminus\{X_i,X_j\}} \prod_{(i,j)\in\mathcal{E}_T} f_{ij}(X_i, X_j) \\
&= \sum_{X\setminus\{X_i,X_j\}} f_{ij}(X_i, X_j) \prod_{(i'j')\in\mathcal{E}_{T_{j\to i}}} f_{i'j'}(X_{i'}, X_{j'}) \prod_{(i'j')\in\mathcal{E}_{T_{i\to j}}} f_{i'j'}(X_{i'}, X_{j'}) \\
&= f_{ij}(X_i, X_j) \left(\sum_{\mathcal{X}_{T_{j\to i}}} \prod_{(i'j')\in\mathcal{E}_{j\to i}} f_{i'j'}(X_{i'}, Y_{j'}) \right) \left(\sum_{\mathcal{X}_{T_{i\to j}}} \prod_{(i'j')\in\mathcal{E}_{T_{i\to j}}} f_{i'j'}(X_{i'}, Y_{j'}) \right) \\
&= f_{ij}(X_i, X_j) \prod_{X_k\in\text{ne}(X_j)\setminus\{X_i\}} M_{k\to j}(X_j) \prod_{X_k\in\text{ne}(X_i)\setminus\{X_j\}} M_{k\to i}(X_i)
\end{aligned}$$

□

Remark 9. **(Belief Propagation for Inference)** Let's consider the belief propagation for inference on the following graphical model:



To compute the $P(X_i|X_a = a)$ as we have the following message:

$$M_{a\to i} = f_{ai}(X_a = a, X_i)$$

Please note that computing $P(X_i)$ requires that $M_{a\to i} = \sum_{X_a} f_{ai}(X_a, X_i)$. For the internal node that partition the graph, like variable like X_b, we have the following message:

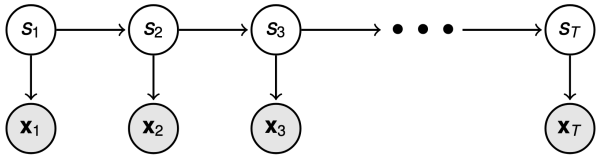
$$M_{b\to j} = f_{bj}(X_b = b, X_j) \qquad M_{b\to k} = f_{bk}(X_b = b, X_k)$$

Please note that $M_{i\to j}$ are proportional to likelihood based on any observed variable (within message subtree) and possibly scaled by the prior (depends on factorization):

$$M_{i\to j}(X_j) \propto P(\mathcal{X}_{T_{i\to j}} \cap \mathcal{O}|X_j)P(X_j)$$

If we consider the message to the observed node, then we have the likelihood. Keeepin all messages unnormalize and any marginal, then the normalizer is the likelihood.

Remark 10. **(BP Latent Chain Model)** We consider the belief propagation in the latent chain model, which is a rooted directed tree, as we have the following graphical model:



We use the backward-forward algorithm is just belief propagation on this graph:

- $\alpha_t(i) = M_{s_{t-1}\to s_t}(s_t = i) \propto P(x_{1:t}, s_t)$
- $\beta_t(i) = M_{s_{t+1}\to s_t}(s_t = i) \propto P(x_{t+1:T}|s_t)$

where we can easily see that:

$$\alpha_t(i)\beta_t(i) = \prod_{j\in \text{ne}(s_t)} M_{j\to s_t}(s_t = i) \propto P(s_t = i|\mathcal{O})$$

The algorithm like BP extend the power of graphical model beyound just encoding of independent and factorization. A single derivation surves multiple models.

2.1 Junction Tree

Remark 11. **(Inference on Graph)** The graphical model sometimes, which is represented by an undirected graph isn't a tree as we would like to find the marginal probability of the single value. There are several strategies that we can use:

- Propagate the local message anyway and hope for the best. This is called "loopy belief propagation", which is an approximation technique.
- Grouping the variable together with multi-variable nodes until the resulting graph, which is a tree as we are going consider this.

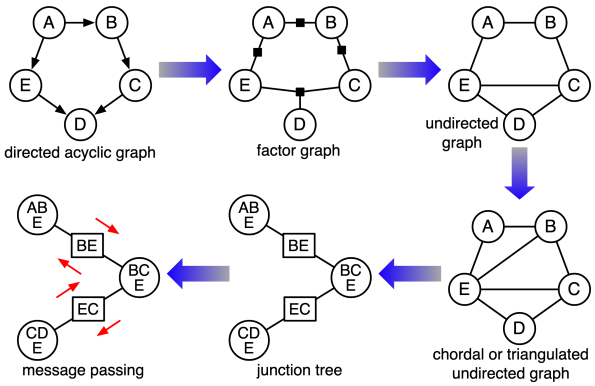
Remark 12. **(Transforming the Graph)** Consider to transform the graph into one that is easier to handle. As the original graph G encodes a distribution $P(\mathcal{X})$ with a certain factorization or independent structure:

- Transformation from G into an easy to handle G' as it will be valid if $P(\mathcal{X})$ can be represented by G'
- Ensuring this, we need every step of the graph transformation only to remove condtional independence (never adding them).
- Making the family of possible encoding distribution groups grows or stay the same at each step.
- The factor potential on the new graph G' are built from those given on G as to make sure it encodes the same distribution.

Definition 2.1. (Junction Tree) A junction tree is a tree whose node and edges are labelled with set of variables.

- Each node is represented by a cliques, edges are labelled by intersection of cliques called seperator.
- The cliques contains all adjacent separator.
- Furthermore, if 2 cliques contain variable X, all cliques and separator on the path between 2 cliques must contain X

Definition 2.2. (Constructing Junction Tree) We consider the following step that transform DAG into a junction tree, which is shown in the figure below:



There are many process that we have to perform.

DAG to Factor Graph: The factor can be see as the conditional distribution of DAG via:

$$P(\mathcal{X}) = \prod_i P(X_i|X_{\text{pa}(i)}) = \prod_i f_i(X_C)$$

where $C_i = i \cup \text{pa}(i)$ and $f_i(X_{C_i}) = P(X_i|X_{\text{pa}(i)})$. Marginal distribution on roots $P(X_r)$ absorbed into an adjacent factor.

Observation in Factor Graph: Usually, inference target a posterior marginal given a set of observed values $P(X_l|\mathcal{O})$; for example, $P(A|D = \text{wet}, C = \text{rain})$. Modify the factor linked to the observed node, or add single factor adjacent to the observed node:

$$f_D(D) = \begin{cases} 1 & \text{if } D = \text{wet} \\ 0 & \text{otherwise} \end{cases} \qquad f_C(C) = \begin{cases} 1 & \text{if } C = \text{rain} \\ 0 & \text{otherwise} \end{cases}$$

Factor Graph to Undirected Graph: The trasnformation from DAG to undirected graph is called moralization, where: Marry all parents of each node by adding an edge to connect them. We also drop arrows on all edges.

Triangulate the Undirected Graph: Want every factor of DAG must be contained within a maximal cliques of the undirected graph. We will have to perform the following modification:

- Replace each factor by an undirected cliques.
- Construct the potential on each maximal clique by multiplying together factor potential that fall on it, and ensure each factor potential only appear once.

To do this, we shall modify the graph as:

- We join the loop into cliques, which can be very inefficient.
- Triangulation is performed to add edges to graph, so that every loop of size ≥ 4 has at least one chord. We will adding it recursively to ensure that the loop ≥ 4 has chords too.
- The graph where every loop of size ≥ 4 has at least one chord is called chordal or triangulated.
- Adding the edge removes conditional independencies, which enlarge the family of distribution.

To find the triangulation is NP-complete problem, so we resort to heuristic called variable elimination. Let's consider the order of elimination as we have:

$$\begin{aligned}
P(X_{(n)}) &= \sum_{X_{\sigma(n-1)}} \cdots \sum_{X_{\sigma(1)}} P(\mathcal{X}) \\
&= \frac{1}{Z} \sum_{X_{\sigma(n-1)}} \cdots \sum_{X_{\sigma(1)}} \prod_i f_i(\mathcal{X}_{C_i}) \\
&= \frac{1}{Z} \sum_{X_{\sigma(n-1)}} \cdots \sum_{X_{\sigma(2)}} \prod_{j:C_j \ni \sigma(2)} f_j(\mathcal{X}_{C_j}) \sum_{X_{\sigma(1)}} \prod_{i:C_i \ni \sigma(1)} f_i(\mathcal{X}_{C_i}) \\
&= \frac{1}{Z} \sum_{X_{\sigma(n-1)}} \cdots \sum_{X_{\sigma(2)}} \prod_{j:C_j \ni \sigma(2)} f_j(\mathcal{X}_{C_j}) f_{\text{new}}(\mathcal{X}_{\text{new}})
\end{aligned}$$

Please note that C_{new} is the neighbour of $X_{\sigma(1)}$ and *edges are added to graph connected all nodes in* C_{new}:

- The graph including of all edges would be induced by elimination is chordal.
- Finding a good triangulation depends on findinga good order of elimination $\sigma(1), \ldots, \sigma(n)$.
- It is NP-complete to find the best heuristic, as there are 2 ways that we pick the next varaible to elimiate as follows:
  - Min-Deficiency Search: Choose variable that induces the fewest new edge.
  - Max-Cardinal Search: Choose node with most previous visited neighbour.

  In most experiments, min-deficiency search seem empirically be better.

Chordal Graph to Junction Tree: To build a junction tree, we follows the procedure as we have:

- Find the maximal clique $C_1, \ldots, C_k$ of the chordal undirected graph.
- Create a weighted graph, which nodes are labelled by maximal cliques and edges connected each pair of cliques that shares variables.

- Create an edge with size of separator as we find maximal weight spanning tree of weighted graph.

Thus, we have the junction tree.

Remark 13. **(Junction Tree Figures)** The joint distribution factors over junction tree is:

$$P(\mathcal{X}) = \frac{1}{Z}\prod_i f_i(X_{C_i}) = \dots f_{\text{ABC}}(A,B,C)f_{\text{BCD}}(B,C,D)\dots$$

This violates the usual undirected tree sematics of factor per edge, and so we add the following constriants:

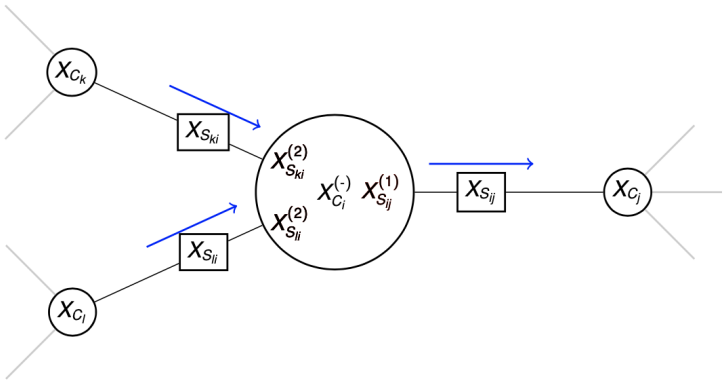
- Introducing the copy of the same variable so that there is no overlaps.
- Adding new delta function that enforce consistency:

$$P(\mathcal{X}) = \dots f_{\text{ABC}}(A,B^{(1)},C^{(1)})\underbrace{\delta(B^{(1)}-B^{(2)})\delta(C^{(1)},C^{(2)})}_{f_{\text{sep}}(B^{(1)},C^{(1)},B^{(2)},C^{(2)})} f_{\text{BCD}}(B^{(2)},C^{(2)},D)\dots$$

Having a new message passing the junction to be:

- Unshared Variable $X_{C_i}^{(-)} = X_{C_i \setminus \bigcup S_{ik}}$
- Variable incoming separator $X_{S_{ki}}^{(2)}$ (Same as matching variable $X_{S_{ki}}^{(1)}$ in $k \in \text{ne}(i)\setminus j$)
- Variable outgoing separator $X_{S_{ij}}^{(1)}$ (Same as matching varaible $X_{S_{ij}}^{(2)}$ in clique j)
- The variable that appear in more than one separator will need additional copies.

The overall process is shown in the following junction tree figure:



Definition 2.3. (Shafer-Shenopy) We have the following formula of the message:

$$\begin{aligned} M_{i\to j}(X_{S_{ij}}^{(2)}) &= \sum_{X_{C_i}^{(-)},\left\{X_{S_{ki}}^{(2)}\right\},S_{S_{ij}}^{(+)}} f_i\left(X_{C_i}^{(-)},\left\{X_{S_{ki}}^{(2)}\right\},S_{S_{ij}}^{(+)}\right) f_{ij}\left(X_{S_{ij}}^{(1)},X_{S_{ij}}^{(2)}\right)\prod_{k\in\text{ne}(i)\setminus j} M_{k\to i}(X_{S_{ij}}^{(2)}) \\ &= \sum_{X_{C_i}\setminus S_{ij}} f_i(X_{C_i})\prod_{k\in\text{ne}(i)\setminus j} M_{k\to i}(X_{S_{ij}}) \end{aligned}$$

The marginal distribution on cliques and separator are defined by:

$$P(X_{C_i}) \propto f_i(X_{C_i})\prod_{k\in\text{ne}(C_i)} M_{k\to i}(X_{S_{ki}}) \qquad P(X_{S_{ij}}) \propto M_{i\to j}(X_{S_{ij}})M_{j\to i}(X_{S_{ij}})$$

Remark 14. **(Junction Tree Properties)** The running intersection property and tree structure of the junction tree implies that local consistency between cliques and separator marginal gurantee global consistency: If we consider the distribution $q_i(X_{C_i})$ and $r_{ij}(X_{S_{ij}})$ are distribution such that:

$$\sum_{X_{C_i \setminus S_{ij}}} q_i(X_{C_i}) = r_{ij}(X_{S_{ij}})$$

This implies that that joint distribution to be:

$$P(\mathcal{X}) = \frac{\prod_{\text{cliques } i} q_i(X_{C_i})}{\prod_{\text{separator } (ij)} r_{ij}(X_{S_{ij}})}$$

As we have:

$$q_i(X_{C_i}) = \sum_{\mathcal{X} \setminus X_{C_i}} P(\mathcal{X}) \qquad r_{ij}(X_{S_{ij}}) = \sum_{\mathcal{X} \setminus X_{S_{ij}}} P(\mathcal{X})$$

Definition 2.4. (Hugin Update) Let's start by initializing the variables to be:

$$q_i(X_{C_i}) \propto f_i(X_{C_i}) \qquad r_{ij}(X_{S_{ij}}) \propto 1$$

A Hugin propagation update for $i \to j$ is given by:

$$r_{ij}^{\text{new}} = \sum_{X_{C_i \setminus S_{ij}}} q_i(X_{C_i}) \qquad q_j^{\text{new}}(X_{C_j}) = q_j(X_{C_j}) \frac{r_{ij}^{\text{new}}(X_{S_{ij}})}{r_{ij}(X_{S_{ij}})}$$

Setting the correct marginalization locally for the first update. For the second update, we change the q based on the keeping the joint probability $P(\mathcal{X})$

3 Bayes and Gaussian Processes

3.1 Bayes Method

Remark 15. **(Recap of Bayes)** Model have a parameter θ_m that specify the probability of data $P(\mathcal{D}|\theta_m, m)$. If the model is known, learning θ_m means finding a posterior or point estimate.

- What if we want to learn the type of model too ?
- Can we combine the model into a single "supermodel" with a composite parameters ?
- We can separate the model selection step: $P(\theta_m, m|\mathcal{D}) = P(\theta_m|m, \mathcal{D})P(m|\mathcal{D})$

Definition 3.1. (Neyman-Peason Hypothesis Testing) For nested model, starting with simpliest model with $m = 1$, comparing null hypothesis m to its alternate $m+1$ and repeat until $m+1$ is rejected. Note that this tests only exact when it is asympototic in data number. Finally, it is conservative as it is asymmetric by design.

Definition 3.2. (Likelihood Validation) Partition data into disjoint training and validation set, where $D = D_{\text{tr}} \cup D_{\text{valid}}$. Then, we choose a model with greatest $P(D_{\text{valid}}|\theta_m^{\text{ML}})$ where:

$$\theta_m^{\text{ML}} = \arg\max_{\theta} P(D_{\text{tr}}|\theta)$$

or if we are able to find $P(D_{\text{valid}}|D_{\text{train}}, m)$.

- This model is consistent and select the most useful model even if they are incorrect.

- It may be biased toward a simpler models and gives a high-variance.
- We can use cross-validation is used with multiple partition and average likelihood.

Definition 3.3. (Baysian Model Selection) We would like to choose a model $P(m|\mathcal{D})$ where consistent as it is a probability principle if true model is in the set.

- However, it might be a problem of assumed prior. Finally, the posterior can weights models for combined prediction.
- A model of class m is a set of distributions parameterized by θ_m. The model implies the prior over parameters as the posterior of the parameter:

$$P(\theta_m|\mathcal{D}, m) = \frac{P(\mathcal{D}|\theta_m, m)P(\theta_m|m)}{P(\mathcal{D}|m)} \quad \text{where}$$

  where $P(\mathcal{D}|m)$ is the Baysian evidence for model m, where the ratio is known as Bayes factor:

$$\frac{P(\mathcal{D}|m)}{P(\mathcal{D}|m')} = \frac{P(m|\mathcal{D})}{P(m'|\mathcal{D})}\frac{P(m')}{P(m)}$$

- This is linked to Occam's razor where: the model that are *too complex* can generate many data but they can be unlikely to generate a particular dataset. The model that are *too simple* can't generate the data.

Remark 16. **(Conjugate Prior)** We will recall the use of exponential model. Suppose, we have $P(\mathcal{D}|\boldsymbol{\theta}_m, m)$ is member of the exponential family, where we can see that:

$$P(\mathcal{D}|\boldsymbol{\theta}_m, m) = \prod_{i=1}^{N} P(\boldsymbol{x}_i|\boldsymbol{\theta}_m, m) = \prod_{i=1}^{N} \exp\left(\boldsymbol{s}(\boldsymbol{x}_i)^T\boldsymbol{\theta}_m - A(\boldsymbol{\theta}_m)\right)$$

Please note that $\boldsymbol{A}(\boldsymbol{\theta}_m)$ is the normalizing factor as it is (recall exponential family of the form in previous course) equal to $\ln(g(\boldsymbol{\theta}_m))$. Consider the prior conjugate to be:

$$P(\boldsymbol{\theta}|m) = \frac{1}{Z(s_p, n_p)} \exp\left(\boldsymbol{s}_p^T\boldsymbol{\theta}_m - n_p A(\boldsymbol{\theta}_m)\right)$$

Then the posterior is equal to:

$$P(\mathcal{D}, \boldsymbol{\theta}_m|m) = \frac{1}{Z(\boldsymbol{s}_p, p)} \exp\left(\left(\sum_{i=1}^{N} \boldsymbol{s}(\boldsymbol{x}_i) + \boldsymbol{s}_p\right)^T \boldsymbol{\theta}_m - (N + n_p)A(\boldsymbol{\theta}_m)\right)$$

One can show that:

$$P(\mathcal{D}|m) = \int P(\mathcal{D}, \boldsymbol{\theta}_m|m)\ \mathrm{d}\boldsymbol{\theta}_m = \frac{Z(\sum_i \boldsymbol{s}(\boldsymbol{x}_i) + \boldsymbol{s}_p, N + n_p)}{Z(\boldsymbol{s}_p, p)}$$

Remark 17. **(Laplace Approximation)** To find $P(\mathcal{D}|m)$, we will have to perform the following integration:

$$P(\mathcal{D}|m) = \int P(\mathcal{D}, \boldsymbol{\theta}_m|m)\ \mathrm{d}\boldsymbol{\theta}_m$$

as the datasize N grows, $\boldsymbol{\theta}$ becomes more concentrated as $P(\mathcal{D}, \boldsymbol{\theta}_m|m) \propto P(\boldsymbol{\theta}_m|\mathcal{D}, m)$ becomes concentrated

on $\boldsymbol{\theta}_m^*$. Let's try to approximate the $P(\mathcal{D}, \boldsymbol{\theta}_m|m)$ to the second order around $\boldsymbol{\theta}_m^*$:

$$\begin{aligned}
\int P(\mathcal{D}, \boldsymbol{\theta}_m|m)\ \mathrm{d}\boldsymbol{\theta}_m &= \int \exp(\log P(\mathcal{D}, \boldsymbol{\theta}_m|m))\ \mathrm{d}\boldsymbol{\theta}_m \\
&\approx \int \exp\Big[\log P(\mathcal{D}, \boldsymbol{\theta}_m^*|m) + \underbrace{\nabla \log P(\mathcal{D}, \boldsymbol{\theta}_m^*|m)(\boldsymbol{\theta}_m - \boldsymbol{\theta}_m^*)}_{0} \\
&\qquad + \frac{1}{2}(\boldsymbol{\theta}_m - \boldsymbol{\theta}_m^*)^T \underbrace{\nabla^2 \log P(\mathcal{D}, \boldsymbol{\theta}_m^*|m)}_{-\boldsymbol{A}}(\boldsymbol{\theta}_m - \boldsymbol{\theta}_m^*)\Big]\ \mathrm{d}\boldsymbol{\theta}_m \\
&= \int P(\mathcal{D}, \boldsymbol{\theta}_m^*|m) \exp\left[-\frac{1}{2}(\boldsymbol{\theta}_m - \boldsymbol{\theta}_m^*)^T \boldsymbol{A}(\boldsymbol{\theta}_m - \boldsymbol{\theta}_m^*)\right]\ \mathrm{d}\boldsymbol{\theta} \\
&= P(\mathcal{D}|\boldsymbol{\theta}_m^*, m) P(\boldsymbol{\theta}_m^*|m)(2\pi)^{-d/2}\, |\boldsymbol{A}|^{-1/2}
\end{aligned}$$

This is approximating the posterior by a Gaussian, where an approximate that is asymmetrically correct.

Definition 3.4. (Baysian Information Criterion) BIC can be obtained from Laplace approximate:

$$\log P(\mathcal{D}|m) \approx \log P(\boldsymbol{\theta}^*|m) + \log P(\mathcal{D}|\boldsymbol{\theta}_m^*, m) + \frac{d}{2}\log 2\pi - \frac{1}{2}\log|\boldsymbol{A}|$$

where we further have:

$$\boldsymbol{A} = -\nabla^2 \log P(\mathcal{D}, \boldsymbol{\theta}_m^*|m) = -\nabla^2 \log P(\mathcal{D}|\boldsymbol{\theta}^*, m) - \nabla^2 \log P(\boldsymbol{\theta}^*|m)$$

As $N = |D| \to \infty$ and $A \to N\boldsymbol{A}_0 + \text{const}$ for a fixed positive definite matrix $\boldsymbol{A}_0 = \left\langle -\nabla^2 \log P(\boldsymbol{x}|\boldsymbol{\theta}^*, m)\right\rangle$ as we have $\log|N\boldsymbol{A}_0| = d\log N + \log|\boldsymbol{A}_0|$. We will retain only the term that grows with N to be:

$$\log P(\mathcal{D}|m) \approx \log P(\mathcal{D}|\boldsymbol{\theta}_m^*, m) - \frac{d}{2}\log N$$

Remark 18. **(Properties BIC)** Baysian Information Criterion has the following properties, as we have:

- Quick and Easy to compute. It doesn't depend on the prior as we can use ML to estimate instead of MAP estimate.
- It is related to minimal description length. Given the assuption that in large sample limit, all parameter are well determined. But it negated multiple nodes (permuation of mixture of Gaussian).

Definition 3.5. (Hyperparameter and Evidence Optimization) Need to choose between a family of continuous parameterized models:

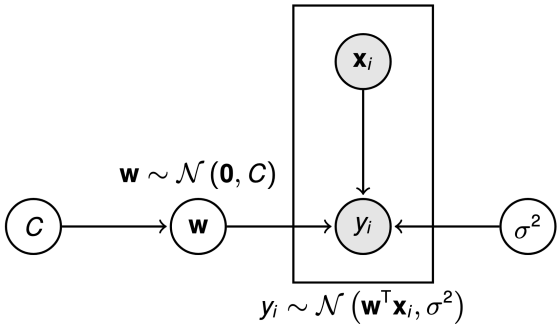
$$P(\mathcal{D}|\boldsymbol{\eta}) = \int P(\mathcal{D}|\boldsymbol{\theta})P(\boldsymbol{\theta}|\boldsymbol{\eta})\ \mathrm{d}\boldsymbol{\theta}$$

We can perform an ascending on gradient in: the exact evidence, approximate evidence (Laplace, BIC) or free-energy based on the evidence (variational Bayes). Performing the hyper-prior on $\boldsymbol{\eta}$, which we can sample its posterior via MCMC as:

$$P(\boldsymbol{\eta}|\mathcal{D}) = \frac{P(\mathcal{D}|\boldsymbol{\eta})P(\boldsymbol{\eta})}{P(\mathcal{D})}$$

3.2 Linear Regression/Gaussian Process

Definition 3.6. (Linear Regression) We consider the following graphical model as:



To find the hyperparameters, we can see that:

$$P(\{y_i\}_{i=1}^N \,|\, \{\boldsymbol{x}_i\}_{i=1}^N, \boldsymbol{C}, \sigma^2) = \int P(\{y_i\}_{i=1}^N \,|\, \{\boldsymbol{x}_i\}_{i=1}^N, \boldsymbol{w}, \sigma^2) P(\boldsymbol{w}|\boldsymbol{C}) \, \mathrm{d}\boldsymbol{w}$$

Maximizing the value of $\boldsymbol{C}$ and σ^2.

Proposition 3.1. *The update rule for the parameter $\boldsymbol{\theta} = \left\{\boldsymbol{C}, \boldsymbol{\sigma}^2\right\}$ is:*

$$\frac{\partial}{\partial \boldsymbol{\theta}} \log \mathcal{E}(\boldsymbol{C}, \sigma^2) = \frac{1}{2} \operatorname{Tr}\left[(\boldsymbol{C} - \boldsymbol{\Sigma}_w - \bar{\boldsymbol{w}}\bar{\boldsymbol{w}}^T) \frac{\partial}{\partial \boldsymbol{\theta}} \boldsymbol{C}^{-1}\right]$$
$$\frac{\partial}{\partial \sigma^2} \log \mathcal{E}(\boldsymbol{C}, \sigma^2) = \frac{1}{\sigma^2}\left(-N + \operatorname{Tr}[\boldsymbol{I} - \boldsymbol{\Sigma}_w \boldsymbol{C}^{-1}] + \frac{1}{\sigma^2}(\boldsymbol{Y} - \boldsymbol{X}^T\bar{\boldsymbol{w}})^T(\boldsymbol{Y} - \boldsymbol{X}^T\bar{\boldsymbol{w}})\right)$$

Proof. We can see that the posterior of $P\left(\boldsymbol{w} \,\middle|\, \{y_i\}_{i=1}^N, \{\boldsymbol{x}_i\}_{i=1}^N, \boldsymbol{C}, \sigma^2\right)$. We can see that the posterior on $\boldsymbol{w}$ is normal with:

$$\Sigma_{\boldsymbol{w}} = \left(\frac{\boldsymbol{X}\boldsymbol{X}^T}{\sigma^2} + \boldsymbol{C}^{-1}\right)^{-1} \qquad \bar{\boldsymbol{w}} = \boldsymbol{\Sigma}_{\boldsymbol{w}} \frac{\boldsymbol{X}\boldsymbol{Y}}{\sigma^2}$$

This follows from the eariler works. Now, we can consider terms inside the exponential of the joint distributions $P(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{w}, \sigma^2)P(\boldsymbol{w}|\boldsymbol{C})$:

$$\begin{aligned}
-\frac{1}{2\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} + \frac{1}{\sigma^2}\boldsymbol{w}^T\boldsymbol{X}\boldsymbol{Y} - \frac{1}{2}\left[\boldsymbol{w}^T\boldsymbol{\Sigma}_w^{-1}\boldsymbol{w}\right] \\
&= -\frac{1}{2\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} - \frac{1}{2}\left\{\boldsymbol{w}^T\boldsymbol{\Sigma}_w^{-1}\boldsymbol{w}^T - \frac{2}{\sigma^2}\boldsymbol{w}^T\boldsymbol{\Sigma}_w\boldsymbol{\Sigma}_w^{-1}\boldsymbol{X}\boldsymbol{Y}\right\} \\
&= -\frac{1}{2\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} - \frac{1}{2}\left\{\boldsymbol{w}^T\boldsymbol{\Sigma}_w^{-1}\boldsymbol{w}^T - \frac{2}{\sigma^2}\boldsymbol{w}^T\boldsymbol{\Sigma}_w\boldsymbol{\Sigma}_w^{-1}\boldsymbol{X}\boldsymbol{Y} + \frac{1}{\sigma^4}\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w^T\boldsymbol{\Sigma}_w^{-1}\boldsymbol{\Sigma}_w^T\boldsymbol{X}\boldsymbol{Y}\right\} \\
&\qquad + \frac{1}{2\sigma^4}\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w^T\boldsymbol{X}\boldsymbol{Y} \\
&= -\frac{1}{2\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} + \frac{1}{2\sigma^4}\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w^T\boldsymbol{X}\boldsymbol{Y} - \frac{1}{2}\left(\boldsymbol{w} - \frac{1}{\sigma^2}\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}\right)^T\boldsymbol{\Sigma}_w^{-1}\left(\boldsymbol{w} - \frac{1}{\sigma^2}\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}\right)
\end{aligned}$$

If we consider the quadratic terms, and the integration over it we have:

$$\int \exp\left\{-\frac{1}{2}\left(\boldsymbol{w} - \frac{1}{\sigma^2}\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}\right)^T\boldsymbol{\Sigma}_w^{-1}\left(\boldsymbol{w} - \frac{1}{\sigma^2}\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}\right)\right\} \mathrm{d}\boldsymbol{w}$$

This is the Gaussian integration, which it is a constant that we can ignore. This leads to the evidence to be (the normalizing factor can be easily found):

$$\mathcal{E}(\boldsymbol{C}, \sigma^2) = \sqrt{\frac{|2\pi\boldsymbol{\Sigma}_w|}{|2\pi\sigma^2\boldsymbol{I}|\,|2\pi\boldsymbol{C}|}} \exp\left\{-\frac{1}{2}\boldsymbol{Y}^T\left(\frac{I}{\sigma^2} - \frac{\boldsymbol{X}^T\boldsymbol{\Sigma}_w\boldsymbol{X}}{\sigma^4}\right)\boldsymbol{Y}\right\}$$

Now, let's consider the derivative to be, starting with the value of $\boldsymbol{C}$, we can see that the relevant values:

$$
\begin{aligned}
&\frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}}\log|\boldsymbol{\Sigma}_w| - \frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}}\log|\boldsymbol{C}| + \frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}}\frac{\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}}{\sigma^4} \\
&\quad= -\frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}}\left|\boldsymbol{\Sigma}_w^{-1}\right| + \frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}}\log\left|\boldsymbol{C}^{-1}\right| + \frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}}\frac{\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}}{\sigma^4} \\
&\quad= -\frac{1}{2}\operatorname{Tr}\left(\Sigma_w\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) + \frac{1}{2}\operatorname{Tr}\left(\boldsymbol{C}\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) + \frac{1}{2\sigma^4}\operatorname{Tr}\left(\boldsymbol{X}\boldsymbol{Y}\boldsymbol{Y}^T\boldsymbol{X}^T\frac{\partial \boldsymbol{\Sigma}_w^{-1}}{\partial \boldsymbol{\theta}}\right) \\
&\quad= -\frac{1}{2}\operatorname{Tr}\left(\Sigma_w\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) + \frac{1}{2}\operatorname{Tr}\left(\boldsymbol{C}\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) \\
&\qquad\qquad - \frac{1}{2\sigma^4}\operatorname{Tr}\left(\boldsymbol{X}\boldsymbol{Y}\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w^{-1}\frac{\partial}{\partial \boldsymbol{\theta}}\left[\frac{\boldsymbol{X}\boldsymbol{X}^T}{\sigma^2} + \boldsymbol{C}^{-1}\right]\boldsymbol{\Sigma}_w\right) \\
&\quad= -\frac{1}{2}\operatorname{Tr}\left(\Sigma_w\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) + \frac{1}{2}\operatorname{Tr}\left(\boldsymbol{C}\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) \\
&\qquad\qquad - \frac{1}{2\sigma^4}\operatorname{Tr}\left(\boldsymbol{X}\boldsymbol{Y}\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\left[\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right]\boldsymbol{\Sigma}_w\right) \\
&\quad= -\frac{1}{2}\operatorname{Tr}\left(\Sigma_w\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) + \frac{1}{2}\operatorname{Tr}\left(\boldsymbol{C}\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) \\
&\qquad\qquad - \frac{1}{2}\operatorname{Tr}\left(\frac{\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w}{\sigma^2}\left[\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right]\frac{\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}}{\sigma^2}\right) \\
&\quad= -\frac{1}{2}\operatorname{Tr}\left(\Sigma_w\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) + \frac{1}{2}\operatorname{Tr}\left(\boldsymbol{C}\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right) - \frac{1}{2}\operatorname{Tr}\left(\bar{\boldsymbol{w}}\bar{\boldsymbol{w}}^T\left[\frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{\theta}}\right]\right)
\end{aligned}
$$

Please recall the derivative of the inverse, and we finish the proof. Now consider the derivative of the evidence with respected to $S = \sigma^2$,

$$
\underbrace{\frac{1}{2}\frac{\partial}{\partial S}\log|\boldsymbol{\Sigma}_w|}_{①} - \frac{N}{2}\frac{\partial}{\partial S}\log S - \frac{1}{2}\frac{\partial}{\partial S}\frac{\boldsymbol{Y}^T\boldsymbol{Y}}{S} + \underbrace{\frac{1}{2}\frac{\partial}{\partial S}\frac{\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}}{S^2}}_{②}
$$

And we have the following derivative for ①, as we have:

$$
\begin{aligned}
\frac{\partial}{\partial S}\log|\boldsymbol{\Sigma}_w| &= -\frac{\partial}{\partial S}\log\left|\boldsymbol{\Sigma}_w^{-1}\right| \\
&= -\operatorname{Tr}\left(\boldsymbol{\Sigma}_w\frac{\partial \boldsymbol{\Sigma}_w^{-1}}{\partial S}\right) \\
&= -\operatorname{Tr}\left(\boldsymbol{\Sigma}_w\frac{\partial}{\partial S}\left[\frac{\boldsymbol{X}\boldsymbol{X}^T}{S} + \boldsymbol{C}^{-1}\right]\right) \\
&= \frac{1}{S^2}\operatorname{Tr}\left(\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{X}^T\right)
\end{aligned}
$$

For for the equation ②, we have:

$$
\begin{aligned}
\frac{\partial}{\partial S}\frac{\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}}{S^2} &= \boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}\frac{\partial}{\partial S}\frac{1}{S^2} + \frac{1}{S^2}\frac{\partial}{\partial S}\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y} \\
&= -2\frac{\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y}}{S^3} + \frac{1}{S}\boldsymbol{Y}^T\boldsymbol{X}^T\boldsymbol{\Sigma}_w\frac{\partial}{\partial S}\left[\frac{\boldsymbol{X}\boldsymbol{X}^T}{S} + C^{-1}\right]\frac{1}{S}\boldsymbol{\Sigma}_w\boldsymbol{X}\boldsymbol{Y} \\
&= -2\boldsymbol{Y}^T\boldsymbol{X}^T\bar{\boldsymbol{w}} + \bar{\boldsymbol{w}}^T\left(\boldsymbol{X}\boldsymbol{X}^T\right)\bar{\boldsymbol{w}}
\end{aligned}
$$

Combining them and we have:

$$\frac{1}{2S}\left(\frac{1}{S}\operatorname{Tr}\left(\boldsymbol{\Sigma}_w \boldsymbol{X}\boldsymbol{X}^T\right) - N + \frac{1}{S}\boldsymbol{Y}^T\boldsymbol{Y} - \frac{1}{S}2\boldsymbol{Y}^T\boldsymbol{X}^T\bar{\boldsymbol{w}} + \frac{1}{S}\bar{\boldsymbol{w}}^T\Sigma_w\left(\boldsymbol{X}\boldsymbol{X}^T\right)\bar{\boldsymbol{w}}\right)$$
$$= \frac{1}{2S}\left(-N + \frac{1}{S}\operatorname{Tr}\left(\boldsymbol{\Sigma}_w \boldsymbol{X}\boldsymbol{X}^T\right) + \frac{1}{S}(\boldsymbol{Y} - \boldsymbol{X}^T\bar{\boldsymbol{w}})^T(\boldsymbol{Y} - \boldsymbol{X}^T\bar{\boldsymbol{w}})\right)$$

Please note that there are some differeces between this and the final proof, which we can't get it but most of them seem correct already □

Definition 3.7. (Automatic Relevence Determination (ARD)) The most common form of evidence optimization with $\boldsymbol{C}^{-1} = \operatorname{diag}(\boldsymbol{\alpha})$ and the optimize $\{\alpha_i\}$ setting the gradient to 0, which we now have:

$$\alpha_i^{\text{new}} = \frac{1 - \alpha_i[\boldsymbol{\Sigma}_n]_{ii}}{\bar{w}_i^2} \qquad (\alpha^2)^{\text{new}} = \frac{(\boldsymbol{Y} - \boldsymbol{X}^T\bar{\boldsymbol{w}})^T(\boldsymbol{Y} - \boldsymbol{X}^T\bar{\boldsymbol{w}})}{N - \sum_i(1 - [\boldsymbol{\Sigma}_w]_{ii}\alpha_i)}$$

During the optimization, there are 2 possible scenario for $\boldsymbol{\alpha}$:

- $\alpha_i \to \infty$ where the weight $\alpha_i = 0$
- α_i is finite and so $w_i = \arg\max P(w_i|\boldsymbol{X}, \boldsymbol{Y}, \alpha_i)$

This is called Automatic Relevence Determination (ARD), which yields space solution that improve ML regression (like LASSA) This Evidence maximization is called ML likelihood or ML-2.

Definition 3.8. (Prediction Averaging) We integrate out the parameter, where we have the density estimation:

$$P(\boldsymbol{x}|\mathcal{D}, m) = \int P(\boldsymbol{x}|\boldsymbol{\theta}, m)P(\boldsymbol{\theta}|\mathcal{D}, m)\ \mathrm{d}\boldsymbol{\theta}$$

Or, perform the prediction:

$$P(y|\boldsymbol{x}, \mathcal{D}, m) = \int P(y|\boldsymbol{x}, \theta, m)P(\theta|\mathcal{D}, m)\ \mathrm{d}\boldsymbol{\theta}$$

This kind of prediction might resists overfitting with infintely complex model as this can be called Baysian non-parametric, which leads to Gaussian Process.

Proposition 3.2. ***(Prediction)*** *Given a new input vector $\boldsymbol{x}$, the predicted output y is given by:*

$$y|\boldsymbol{x} \sim \mathcal{N}(\bar{\boldsymbol{w}}^T\boldsymbol{x}, \boldsymbol{x}^T\boldsymbol{\Sigma}_w\boldsymbol{x} + \sigma^2)$$

This comes from the linear Gaussian model. The variance $\boldsymbol{x}^T\boldsymbol{\Sigma}_w\boldsymbol{x}$ that comes from the uncertainty of $\boldsymbol{w}$.

Remark 19. **(Observation on Joint Distribution)** Consider the joint probability between $y_1, \ldots, y_N$ and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, and noisy prediction with variance τ^2, we can see that the mean and the covariances are:

$$\mathbb{E}[y_i] = \mathbb{E}[\boldsymbol{w}^T\boldsymbol{x}_i] = 0$$
$$\mathbb{E}[(y_i - \bar{y}_i)^2] = \mathbb{E}[(\boldsymbol{x}_i^T\boldsymbol{w})(\boldsymbol{w}^T\boldsymbol{x}_i)] + \sigma^2 = \tau^2\boldsymbol{x}_i^T\boldsymbol{x}_i + \sigma^2$$
$$\mathbb{E}[(y_i - \bar{y}_i)(y_j - \bar{y}_j)] = \mathbb{E}[(\boldsymbol{x}_i^T\boldsymbol{w})(\boldsymbol{w}^T\boldsymbol{x}_j)] = \tau^2\boldsymbol{x}_i^T\boldsymbol{x}_j$$

So the leads to the following joint distributions:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \middle| \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} \tau^2\boldsymbol{x}_1^T\boldsymbol{x}_1 + \sigma^2 & \tau^2\boldsymbol{x}_1^T\boldsymbol{x}_2 & \cdots & \tau^2\boldsymbol{x}_1^T\boldsymbol{x}_N \\ \tau^2\boldsymbol{x}_2^T\boldsymbol{x}_1 & \tau^2\boldsymbol{x}_2^T\boldsymbol{x}_2 + \sigma^2 & \cdots & \tau^2\boldsymbol{x}_2^T\boldsymbol{x}_N \\ \vdots & \vdots & \ddots & \vdots \\ \tau^2\boldsymbol{x}_N^T\boldsymbol{x}_1 & \tau^2\boldsymbol{x}_N^T\boldsymbol{x}_2 & \cdots & \tau^2\boldsymbol{x}_N^T\boldsymbol{x}_N + \sigma^2 \end{bmatrix}\right)$$

If we were to consider adding the test input vector $\boldsymbol{x}$ and test vector y, we have:

$$\begin{bmatrix}\boldsymbol{Y}^T\\ y\end{bmatrix}\Big|\,\boldsymbol{X},\boldsymbol{x}\sim\mathcal{N}\left(\begin{bmatrix}\boldsymbol{0}\\0\end{bmatrix},\begin{bmatrix}\tau\boldsymbol{X}^T\boldsymbol{X}+\sigma^2\boldsymbol{I} & \tau^2\boldsymbol{X}^T\boldsymbol{x}\\ \tau^2\boldsymbol{x}^T\boldsymbol{X} & \tau\boldsymbol{x}^T\boldsymbol{x}+\sigma^2\end{bmatrix}\right)=\mathcal{N}\left(\begin{bmatrix}\boldsymbol{0}\\0\end{bmatrix},\begin{bmatrix}\tilde{\boldsymbol{K}}_{XX} & \tilde{\boldsymbol{K}}_{Xx}\\ \tilde{\boldsymbol{K}}_{xX} & \tilde{\boldsymbol{K}}_{xx}\end{bmatrix}\right)$$

By the linear Gaussian model, we are given, the following conditional

$$\begin{aligned}y|\boldsymbol{Y},\boldsymbol{X},\boldsymbol{x}&\sim\mathcal{N}\Big(\tilde{\boldsymbol{K}}_{xX}\tilde{\boldsymbol{K}}_{XX}^{-1}\boldsymbol{Y}^T,\ \tilde{\boldsymbol{K}}_{xx}-\tilde{\boldsymbol{K}}_{xX}\tilde{\boldsymbol{K}}_{XX}^{-1}\tilde{\boldsymbol{K}}_{Xx}\Big)\\&\sim\mathcal{N}\Big(\tau^2\boldsymbol{x}^T\boldsymbol{X}(\tau^2\boldsymbol{X}^T\boldsymbol{X}+\sigma^2\boldsymbol{I})^{-1}\boldsymbol{Y}^T,\ \tau^2\boldsymbol{x}^T\boldsymbol{x}+\sigma^2-\tau^2\boldsymbol{x}^T\boldsymbol{X}(\tau^2\boldsymbol{X}^T\boldsymbol{X}+\sigma^2)^{-1}\tau^2\boldsymbol{X}^T\boldsymbol{x}\Big)\\&\sim\mathcal{N}\left(\boldsymbol{x}^T\frac{1}{\sigma^2}\boldsymbol{\Sigma}\boldsymbol{X}\boldsymbol{Y}^T,\boldsymbol{x}^T\boldsymbol{\Sigma}\boldsymbol{x}+\sigma^2\right)\end{aligned}$$

where $\boldsymbol{\Sigma}=(\frac{1}{\sigma^2}\boldsymbol{X}\boldsymbol{X}^T+\frac{1}{\tau^2}\boldsymbol{I})^{-1}$. The derivation comes from the Woodbury identity.

Remark 20. **(Non-Linear Regression)** We consider the mapping $\boldsymbol{x}\mapsto\boldsymbol{\phi}(\boldsymbol{x})$, the regression:

$$\boldsymbol{Y}|\boldsymbol{X}\sim\mathcal{N}(\boldsymbol{0}_N,\tau^2\boldsymbol{\Phi}^T\boldsymbol{\Phi}+\sigma^2\boldsymbol{I}_N)$$

where i-th column of matrix $\boldsymbol{\Phi}$ is $\boldsymbol{\phi}(\boldsymbol{x}_i)$. Now we have the following prediction:

$$\boldsymbol{y}|\boldsymbol{Y},\boldsymbol{X},\boldsymbol{x}\sim\mathcal{N}(\tilde{\boldsymbol{K}}_{xX}\tilde{\boldsymbol{K}}_{XX}^{-1}\boldsymbol{Y}^T,\ \tilde{\boldsymbol{K}}_{xx}-\tilde{\boldsymbol{K}}_{xX}\tilde{\boldsymbol{K}}_{XX}^{-1}\tilde{\boldsymbol{K}}_{Xx})$$

where we have:

$$\tilde{\boldsymbol{K}}_{XX}=\tau^2\boldsymbol{\Phi}^T\boldsymbol{\Phi}+\sigma^2\boldsymbol{I}\qquad\tilde{\boldsymbol{K}}_{Xx}=\tau^2\boldsymbol{\Phi}^T\boldsymbol{\phi}(\boldsymbol{x})\qquad\tilde{\boldsymbol{K}}_{xx}=\tau^2\boldsymbol{\phi}(\boldsymbol{x})^T\boldsymbol{\phi}(\boldsymbol{x})+\sigma^2$$

Remark 21. **(Kernel Vector)** Define a covariance kernel function $\tilde{K}:\mathbb{X}\times\mathbb{X}\to\mathbb{R}$ if $\boldsymbol{x},\boldsymbol{x}'\in\mathbb{X}$ are 2 inputs vectors with corresponding output y and y', then we have:

$$\tilde{K}(\boldsymbol{x},\boldsymbol{x}')=\mathrm{Cov}[y,y']=\mathbb{E}[yy']-\mathbb{E}[y]\mathbb{E}[y']$$

In non-linear regression problem we have $\tilde{K}(\boldsymbol{x},\boldsymbol{x}')=\tau^2\boldsymbol{\phi}(\boldsymbol{x})^T\boldsymbol{\phi}(\boldsymbol{x}')+\sigma^2\delta[\boldsymbol{x}=\boldsymbol{x}']$, where the prediction depends on $\tilde{\boldsymbol{K}}(\boldsymbol{x},\boldsymbol{x}')$ rather than its feature map $\boldsymbol{\phi}(\boldsymbol{x})$ and so, we have:

$$\boldsymbol{Y}|\boldsymbol{X},\tilde{\boldsymbol{K}}\sim\mathcal{N}(\boldsymbol{0}_N,\tilde{\boldsymbol{K}}_{XX})$$

To perform a prediction, as we have:

$$\boldsymbol{y}|\boldsymbol{Y},\boldsymbol{X},\boldsymbol{x}\sim\mathcal{N}(\tilde{\boldsymbol{K}}_{xX}\tilde{\boldsymbol{K}}_{XX}^{-1}\boldsymbol{Y}^T,\ \tilde{\boldsymbol{K}}_{xx}-\tilde{\boldsymbol{K}}_{xX}\tilde{\boldsymbol{K}}_{XX}^{-1}\tilde{\boldsymbol{K}}_{Xx})$$

where $\left[\tilde{\boldsymbol{K}}_{XX}\right]_{ij}=\tilde{\boldsymbol{K}}(\boldsymbol{x}_i,\boldsymbol{x}_j)$, $\left[\tilde{\boldsymbol{K}}_{Xx}\right]_i=\tilde{\boldsymbol{K}}(\boldsymbol{x}_i,\boldsymbol{X})$ and $\tilde{K}_{xx}=\tilde{\boldsymbol{K}}(\boldsymbol{x},\boldsymbol{x})$, now we can have the kernel function solely without the use of the feature map.

Definition 3.9. (Gaussian Process) Let $f(\boldsymbol{x})$ be random variable indexed by $\boldsymbol{x}$, then drawn from the whole GP, which is a random function $f:\mathbb{X}\to\mathbb{R}$ where:

$$f(\cdot)\sim\mathcal{GP}(m(\cdot),k(\cdot,\cdot))$$

It is defiend such that the finite list of points $\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N\}$. The joint distributions of the function value $f=[f(\boldsymbol{x}_1),\ldots,f(\boldsymbol{x}_N)]^T$ as we have $\boldsymbol{f}|\boldsymbol{X},\boldsymbol{K}\sim\mathcal{N}(\boldsymbol{m},\boldsymbol{K}_{XX})$ with the following parameters:

$$[\boldsymbol{K}_{XX}]_{ij}=K(\boldsymbol{x}_i,\boldsymbol{x}_j)\qquad[\boldsymbol{m}]_i=\boldsymbol{m}(\boldsymbol{x}_i)$$

Remark 22. **(Properties of Gaussian Process + Bayes Theorem)** If we assume the random function is dranw from GP prior $f(\cdot)\sim\mathcal{GP}(\boldsymbol{0},K(\cdot,\cdot))$. Observation y_i taken to be noisy version of latent function $f(\boldsymbol{x}_i)$ where we have:

$$\boldsymbol{y}_i|\boldsymbol{x}_i,f(\cdot)\sim\mathcal{N}(y(\boldsymbol{x}_i),\sigma^2)$$

The we have the following quantities:

- The latent f is also Gaussian process:

$$f(\cdot)|\boldsymbol{X},\boldsymbol{Y} \sim \mathcal{GP}\Big(\boldsymbol{K}_{\cdot X}(\boldsymbol{K}_{XX}+\sigma^2 I)^{-1}\boldsymbol{Y}^T,\ K(\cdot,\cdot)-K_{\cdot X}(\boldsymbol{K}_{XX}+\sigma^2\boldsymbol{I})^{-1}\boldsymbol{K}_{X\cdot}\Big)$$

- This is given multivariate Gaussian likelihood:

$$P(\boldsymbol{Y}|\boldsymbol{X}) = \Big|2\pi(\boldsymbol{K}_{xx}+\sigma^2\boldsymbol{I})^{-1/2}\Big| \exp\left(-\frac{1}{2}\boldsymbol{Y}(\boldsymbol{K}_{xx}+\sigma^2\boldsymbol{I})\boldsymbol{Y}^T\right)$$

- The posterior of f with observation noise, as we have:

$$y|\boldsymbol{X},\boldsymbol{Y},\boldsymbol{x} \sim \mathcal{N}(\mathbb{E}[f(\boldsymbol{x})|\boldsymbol{X},\boldsymbol{Y}], \mathrm{Var}[f(\boldsymbol{x})|\boldsymbol{X},\boldsymbol{Y}]+\sigma^2)$$

4 Factored Variation Approximation

Remark 23. **(Intractable EM-Method)** Sometime E-step or M-step are intractable; therefore, we will have to do some approximation to the problem:

- It might be the case that $P(\mathcal{Z}|\mathcal{X},\theta)$ and so we might replace M-step with gradient M-step, where every step increase the likelihood.
- In E-step, we should consider parameterized $q = q(\mathcal{Z})$ and take the gradient step in ρ.
- Or we assume some simplification for q usually $q = \prod_i q_i(\mathcal{Z}_i)$ where $\mathcal{Z}_i$ is the partition $\mathcal{Z}$.

Remark 24. **(Restricting Variational Class)** This approximation of E-step is the same as choosing q from the limit set $\mathcal{Q}$ such that VE-step is to maximize $\mathcal{F}(q,\boldsymbol{\theta})$ such that:

$$q^{(k)}(\mathcal{Z}) = \arg\max_{q\in\mathcal{Q}} \mathcal{F}(q(\mathcal{F}), \boldsymbol{\theta}^{(k-1)})$$

The effect of the restricting q to $\mathcal{Q}$, there is the following effect:

- The free energy is bounded by log-likelihood via Jensen's inequality as $\mathcal{F}(q,\boldsymbol{\theta}) \le l(\theta^{\mathrm{ML}})$, thus as long as every step increases $\mathcal{F}$ convergence still guarantee.
- Since $P(\mathcal{Z}|\mathcal{X},\boldsymbol{\theta}^{(k)})$ may not lie in Q and we no longer saturated the bound after E-step. The likelihood may not increase on each full EM step.

Definition 4.1. (Factored Variational E-Step) We consider the following family of distributions:

$$\mathcal{Q} = \left\{ q \,\middle|\, q(\mathcal{Z}) = \prod_i q_i(\mathcal{Z}_i) \right\}$$

where $\mathcal{Z}$ are partitioned into $\{\mathcal{Z}_\rangle\}$. We consider the following maximization of the E-step:

$$q_i^{(k)}(\mathcal{Z}_i) = \arg\max_{q_i(\mathcal{Z}_i)} \mathcal{F}\left(q_i(\mathcal{Z}_i)\prod_{j\neq i} q_j(\mathcal{Z}_j),\ \boldsymbol{\theta}^{(k-1)}\right)$$

Proposition 4.1. *We can show that:*

$$q_i(\mathcal{Z}_i) \propto \exp\left\langle \log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}^{(k-1)})\right\rangle_{\prod_{j\neq i} q_j(\mathcal{Z}_j)}$$

Proof. Given this, the free energy is given by:

$$\begin{aligned}\mathcal{F}\left(q_i(\mathcal{Z}_i)\prod_{j\neq i}q_j(\mathcal{Z}_j),\ \boldsymbol{\theta}^{(k-1)}\right) &= \left\langle \log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}^{(k-1)})\right\rangle_{\prod_j q_j(\mathcal{Z})} + H\left[\prod_j q_j(\mathcal{Z}_j)\right] \\ &= \int q_i(\mathcal{Z}_i)\left\langle \log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}^{(k-1)})\right\rangle_{\prod_{j\neq i} q_j(\mathcal{Z}_j)} \mathrm{d}\mathcal{Z}_i + H[q_i] + \sum_{j\neq i} H[q_j]\end{aligned}$$

We will consider derivative of the Lagragian with normalizer of q_i, then we have:

$$\frac{\partial}{\partial q_i}\left(\mathcal{F} + \lambda\left(\int q_i - 1\right)\right) = \left\langle \log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}^{(k-1)})\right\rangle_{\prod_{j\neq i} q_j(\mathcal{Z}_j)} - \log q_i(\mathcal{Z}_i) - \frac{q_i(\mathcal{Z}_i)}{q_i(\mathcal{Z}_i)} + \lambda = 0$$

This implies that we have the update as defined above. □

Remark 25. **(Observations on Update Rule)** This update of factorized model depends on the expected sufficient stistics of q and this means that we don't need the entire distribution and just relevant expaction. If $\mathcal{Z}_i = z_i$ (factorized over all variables) then this technique is called mean field expectation:

- Suppose $P(\mathcal{X},\mathcal{Z})$ has sufficient statistics that are separated in the latent variables (for example Bolzman machine):

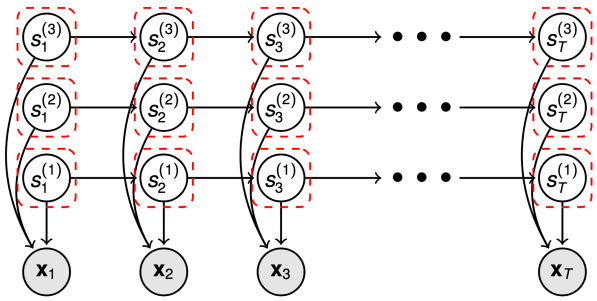
$$P(\mathcal{X},\mathcal{Z}) = \frac{1}{Z}\exp\left(\sum_{ij} W_{ij}s_i s_j + \sum_i b_i s_i\right)$$

- The expectation with respected to fully factored q over $z_i \in \mathcal{Z}$ gives us:

$$\langle \log P(\mathcal{X},\mathcal{Z})\rangle_{\prod_i q_i} = \sum_{ij} W_{ij}\langle s_i\rangle_{q_i}\langle s_j\rangle_{q_j} + \sum_i b_i\langle s_i\rangle_{q_i}$$

The update each q_i given sufficient statistics of the other. Each variables see the mean field imposed by its neighbour and we update these fields until they all agree.

Remark 26. **(Factored Variational Inference in Time Series)** Consider the graphical model:



We consider the following inference with the factorization $\prod_{m,t} q_t^m(s_t^m)$, and so we have:

$$\begin{aligned}q_t^{(m)}\left(s_t^{(m)}\right) &\propto \exp\left\langle \log P\left(s_{1:T}^{1:M}, x_{1:T}\right)\right\rangle_{\prod_{\neg(m,t)} q_{t'}^{m'}(s_{t'}^{m'})} \\ &= \exp\left\langle \sum_{\mu=1}^{M}\sum_{\tau=1}^{T}\log P(s_\tau^\mu|s_{\tau-1}^\mu) + \sum_{\tau=1}^{T}\log P(x_\tau|s_\tau^{1:M})\right\rangle_{\prod_{\neg(m,t)} q_{t'}^{m'}(s_{t'}^{m'})}\end{aligned}$$

We can show that this is propotional to the following:

$$\exp\Big[\underbrace{\left\langle \log P(s_t^m|s_{t-1}^m)\right\rangle_{q_{t-1}^m} + \left\langle \log P(x_t|s_t^{1:M})\right\rangle_{\prod_{\neg m} q_t^{m'}}}_{\alpha_t^m(i)} + \underbrace{\left\langle \log P(s_{t-1}^m|s_t^m)\right\rangle_{q_{t+1}^m}}_{\beta_i^m(i)}\Big]$$

Where we have the following variables (as we have the same message passing algorithm):

$$\alpha_t^m(i) \propto \exp\left[\sum_j \log \Phi_{ji}^m q_{t-1}^m(j)\right] \cdot \exp\left[\langle \log A_i(x_t)\rangle_{q_t^{\neg m}}\right]$$

$$\beta_t^m(i) \propto \exp\left[\sum_j \log \Phi_{ij}^m q_{t+1}^m(j)\right]$$

The update only depends on the immediate neighbours in chain. The chain couple only through joint output. It has multiple passes and message depends on (approximate) marginal. Finally, the evidence doesn't appear explicitly in backward message.

Remark 27. **(Difference Factored Variational Inference)** Consider the difference kind of factorization:

$$q(s_{1:T}^{1:M}) = \prod_m q^m(s_{1:T}^m)$$

This gives us the following graphical model:

$$\begin{aligned} q^{(m)}(s_{1:T}^m) &\propto \exp\left(\left\langle \log p\left(s_{1:T}^{1:M}, x_{1:T}\right)\right\rangle_{\prod_{\neg m} q^{m'}(s_{1:T}^{m'})}\right) \\ &= \exp\left[\sum_{t=1}^T \log P(s_t^m|s_{t-1}^m) + \sum_{t=1}^T \left\langle \log P(x_t|s_t^{1:M})\right\rangle_{\prod_{\neg m} q^{m'}(s_t^{m'})}\right] \end{aligned}$$

Please note that this is similar to standard HMM joint with a modified likelihood term, where we cycle through multiple forward-backward pass updating likelihood terms each time.

Remark 28. **(Message on Arbitrary Graph)** Consider DAG, where we have, the following:

$$P(\mathcal{X}, \mathcal{Z}) = \prod_k P(\mathcal{V}_k|\operatorname{Pa}(\mathcal{V}_k))$$

We let $q(\mathcal{Z}) = \prod_i q_i(\mathcal{Z})$ for disjoint set $\{\mathcal{Z}_i\}$. We have the following VE update:

$$q_i^*(\mathcal{Z}_i) \propto \exp\left(\langle \log P(\mathcal{X}, \mathcal{Z})\rangle_{q_{\neg i}(\mathcal{Z})}\right)$$

Then, we have the following:

$$\begin{aligned} \log q_i^*(\mathcal{Z}_i) &= \left\langle \sum_k P(\mathcal{V}_k|\operatorname{Pa}(\mathcal{V}_k))\right\rangle_{q_{\neg i}(\mathcal{Z})} + \text{const.} \\ &= \sum_{j\in\mathcal{Z}_i} \langle \log P(\mathcal{Z}_j|\operatorname{Pa}(\mathcal{Z}_j))\rangle_{q_{\neg i}(\mathcal{Z})} + \sum_{j\in \operatorname{ch}(\mathcal{Z}_i)} \langle \log P(V_j|\operatorname{Pa}(V_j))\rangle_{q_{\neg i}(\mathcal{Z})} + \text{const.} \end{aligned}$$

Each node receives message from its markov boundary: parent/children/parent of children (all neighbours in the corresponding graphs).

Remark 29. **(Non-Factored Variational Model)** The term variational approximation is used whenever a bound on likelihood is optimized, but not necessary be tight, as we have:

- Closed form update in special case.
- Numerical or sampling based computation of expectation.
- Recognition Network or amortization to approximate variational parameter.
- No free energy based bound on the likelihood.

- We can perform MAP estimate or zero-temperature EM as parametric form of variational infernece.

Proposition 4.2. ***(Variational Bayes)*** *We can consider performing the same for integral over parameter in other to bound the log-marginal likelihood or evidence:*

$$\begin{aligned}\log P(\mathcal{X}|\mathcal{M}) &= \log \iint P(\mathcal{Z},\mathcal{X}|\theta,\mathcal{M})P(\theta|\mathcal{M})\ \mathrm{d}\mathcal{Z}\ \mathrm{d}\theta \\ &= \max_{\mathcal{Q}} \iint Q(\mathcal{Z},\theta)\log\frac{P(\mathcal{X},\mathcal{Z},\theta|\mathcal{M})}{Q(\mathcal{Z},\theta)}\ \mathrm{d}\mathcal{Z}\ \mathrm{d}\theta \\ &\geq \max_{Q_{\mathcal{Z}},Q_{\mathcal{X}}} \iint Q_{\mathcal{Z}}(\mathcal{Z})Q_\theta(\theta)\log\frac{P(\mathcal{X},\mathcal{Z},\theta|\mathcal{M})}{Q_{\mathcal{Z}}(\mathcal{Z})Q_\theta(\theta)}\ \mathrm{d}\mathcal{Z}\ \mathrm{d}\theta = \max_{Q_{\mathcal{Z}},Q_{\mathcal{X}}} \mathcal{F}(Q_{\mathcal{Z}},Q_\theta)\end{aligned}$$

The second equality comes from the Jensen's inequality bound being tight. This leads to variational Bayesian EM algorithm with free energy being $\mathcal{F}(Q_{\mathcal{Z}},Q_\theta)$:

$$\begin{aligned}Q^*_{\mathcal{Z}}(\mathcal{Z}) &\propto \exp\langle\log P(\mathcal{Z},\mathcal{X}|\theta)\rangle_{Q_\theta(\theta)} \\ Q^*_\theta(\theta) &\propto P(\theta)\exp\langle\log(\mathcal{Z},\mathcal{X}|\theta)\rangle_{Q_{\mathcal{Z}}(\theta)}\end{aligned}$$

Proof. The maximizing $\mathcal{F}$ is the same as minimizing KL-divergence between the approximate posterior $Q(\mathcal{Z})Q(\theta)$ and the true posterior $P(\theta,\mathcal{Z}|\mathcal{X})$ as we have:

$$\begin{aligned}\log P(\mathcal{X}) - \mathcal{F}(Q_{\mathcal{Z}},Q_\theta) &= \iint Q_{\mathcal{Z}}(\mathcal{Z})Q_\theta(\theta)\log P(\mathcal{X})\ \mathrm{d}\mathcal{Z}\ \mathrm{d}\theta - \iint Q_{\mathcal{Z}}(\mathcal{Z})Q_\theta(\theta)\log\frac{P(\mathcal{X},\mathcal{Z},\theta|\mathcal{M})}{Q_{\mathcal{Z}}(\mathcal{Z})Q_\theta(\theta)}\ \mathrm{d}\mathcal{Z}\ \mathrm{d}\theta \\ &= \iint Q_{\mathcal{Z}}(\mathcal{Z})Q_\theta(\theta)\log\frac{Q_{\mathcal{Z}}(\mathcal{Z})Q_\theta(\theta)}{P(\mathcal{Z},\theta|\mathcal{X},\mathcal{M})}\ \mathrm{d}\mathcal{Z}\ \mathrm{d}\theta = \mathrm{KL}(Q\|P)\end{aligned}$$

□

Remark 30. **(Conjugate Exponential)** Let's consider the conjugate exponential model on latent variable:

- *Condition 1*: Consider the joint probability over variables in exponential family:

$$P(\mathcal{Z},\mathcal{X}|\boldsymbol{\theta}) = f(\mathcal{X},\mathcal{Z})g(\boldsymbol{\theta})\exp\left\{\boldsymbol{\phi}(\boldsymbol{\theta})^T\boldsymbol{T}(\mathcal{Z},\mathcal{X})\right\}$$

- *Condition 2*: The joint over parameter is conjugate to this joint probability:

$$P(\boldsymbol{\theta}|\nu,\boldsymbol{\tau}) = h(\nu,\boldsymbol{\tau})g(\boldsymbol{\theta})^\nu\exp\left\{\boldsymbol{\phi}(\boldsymbol{\theta})^T\boldsymbol{\tau}\right\}$$

  where ν and $\boldsymbol{\tau}$ on the prior and we can see them as number of pseudo-observations and values of pseudo-observations, respectively.

Remark 31. **(Conjugate Exponential VB)** Given a iid set $\mathcal{D} = \{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_n\}$ if the model is conjugate exponential then we have. $Q_\theta(\boldsymbol{\theta})$ is also conjugate, as we have:

$$\begin{aligned}Q_\theta(\boldsymbol{\theta}) &\propto P(\boldsymbol{\theta})\exp\left\langle\sum_i\log P(\mathcal{Z}_i,\mathcal{X}_i|\boldsymbol{\theta})\right\rangle_{Q_{\mathcal{Z}}(\mathcal{Z})} \\ &= h(\nu,\boldsymbol{\tau})g(\boldsymbol{\theta})^\nu\exp\left(\boldsymbol{\phi}(\boldsymbol{\theta})^T\boldsymbol{\tau}\right)g(\boldsymbol{\theta})^n\exp\left(\left\langle\log f(\mathcal{X},\mathcal{Z}) + \sum_{i=1}^n\boldsymbol{\phi}(\boldsymbol{\theta})^T\boldsymbol{T}(\boldsymbol{z}_i,\boldsymbol{x}_i)\right\rangle_{Q_{\mathcal{Z}}}\right) \\ &\propto h(\tilde{\nu},\tilde{\boldsymbol{\tau}})g(\boldsymbol{\theta})^{\nu+n}\exp\left(\boldsymbol{\phi}(\boldsymbol{\theta})^T\left(\boldsymbol{\tau} + \left\langle\sum_{i=1}^n\boldsymbol{T}(\boldsymbol{z}_i,\boldsymbol{x}_i)\right\rangle_{Q_{\mathcal{Z}}}\right)\right) = h(\tilde{\nu},\tilde{\boldsymbol{\tau}})g(\boldsymbol{\theta})^{\tilde{\nu}}\exp\left(\boldsymbol{\phi}(\boldsymbol{\theta})^T\tilde{\boldsymbol{\tau}}\right)\end{aligned}$$

we can set $\tilde{\nu} = \nu + n$ and $\tilde{\boldsymbol{\tau}} = \tau + \sum_i\langle\boldsymbol{T}(\boldsymbol{z}_i,\boldsymbol{x}_i)\rangle_{Q_{\mathcal{Z}}}$

Remark 32. **(Factorized EM)** We have the following factorized model as we have $Q_{\mathcal{Z}}(\mathcal{Z}) = \prod_{i=1}^{n} Q_{\mathcal{Z}_i}(\mathcal{Z}_i)$ (over the dataset). It has the same form as regular E-step of regular EM, where we have:

$$\begin{aligned} Q_{\mathcal{Z}_i}(\mathcal{Z}_i) &\propto \exp\left(\langle \log P(\boldsymbol{z}_i, \boldsymbol{x}_i)\rangle_{Q_\theta}\right) \\ &\propto f(\boldsymbol{z}_i, \boldsymbol{x}_i) \exp\left\{\langle \boldsymbol{\phi}(\boldsymbol{\theta})\rangle_{Q_\theta}^{T} \boldsymbol{T}(\boldsymbol{z}_i, \boldsymbol{x}_i)\right\} \\ &= P(\boldsymbol{z}_i | \boldsymbol{x}_i, \tilde{\boldsymbol{\phi}}(\boldsymbol{\theta})) \end{aligned}$$

where the natural parameter $\tilde{\boldsymbol{\phi}}(\boldsymbol{\theta}) = \langle \boldsymbol{\phi}(\boldsymbol{\theta})\rangle_{Q_\theta}$, where the inference is unchanged to regular EM.

Remark 33. **(Comparison Between EM for MAP estimation and Variational Baysian EM)** Let's compare the differences between the 2 algorithms:

EM for MAP estimation	**Variational Bayesian EM**
Goal: maximize $P(\boldsymbol{\theta}\|\mathcal{X}, m)$ wrt $\boldsymbol{\theta}$	Goal: maximise bound on $P(\mathcal{X}\|m)$ wrt Q_θ
E Step: compute	**VB-E Step:** compute
$Q_{\mathcal{Z}}(\mathcal{Z}) \leftarrow p(\mathcal{Z}\|\mathcal{X}, \boldsymbol{\theta})$	$Q_{\mathcal{Z}}(\mathcal{Z}) \leftarrow p(\mathcal{Z}\|\mathcal{X}, \bar{\boldsymbol{\phi}})$
M Step:	**VB-M Step:**
$\boldsymbol{\theta} \leftarrow \underset{\boldsymbol{\theta}}{\text{argmax}} \int d\mathcal{Z}\, Q_{\mathcal{Z}}(\mathcal{Z}) \log P(\mathcal{Z}, \mathcal{X}, \boldsymbol{\theta})$	$Q_\theta(\boldsymbol{\theta}) \leftarrow \exp \int d\mathcal{Z}\, Q_{\mathcal{Z}}(\mathcal{Z}) \log P(\mathcal{Z}, \mathcal{X}, \boldsymbol{\theta})$

The following are the feature of VB, as we have:

- It reduces to EM if we set $Q_\theta(\boldsymbol{\theta}) = \delta(\boldsymbol{\theta} - \boldsymbol{\theta}^*)$
- $\mathcal{F}_m$ increase monotonically and interpolate the model complexity penalty.
- We have analytical parameter distribution but we don't have to contraint to Gaussian.
- VB-E step has the same complexity as E-step.
- We can use junction tee, belief propagatio, kalman filter algorithm in VB E-step of VB-EM but we have to use the expected natural parameter $\tilde{\boldsymbol{\phi}}$.

Remark 34. **(VB and Model Classification)** VB-EM gives us the approximate posterior Q_θ over the model parameter:

- It also gives us the lower bound on the model estimate as we have:

$$\max \mathcal{F}_M(Q_{\mathcal{Z}}, Q_\theta) \leq P(\mathcal{D}|\mathcal{M})$$

- These lower bound can be compared amongts model to find the right out. However, if we consider continuous domain of model that is specified by the hyperparameter η, then VB free energy depends on parameter

$$\mathcal{F}(Q_{\mathcal{Z}}, Q_\theta, \eta) = \iint Q_{\mathcal{Z}}(\mathcal{Z}) Q_\theta(\boldsymbol{\theta}) \log \frac{P(\mathcal{X}, \mathcal{Z}, \boldsymbol{\theta}|\eta)}{Q_{\mathcal{Z}}(\mathcal{Z}) Q_\theta(\boldsymbol{\theta})} \, \mathrm{d}\mathcal{Z} \, \mathrm{d}\boldsymbol{\theta} \leq P(\mathcal{X}|\eta)$$

  Hyperparameter-M step maximizes the current bound with respected to η:

$$\eta \leftarrow \arg\max_{\eta} \iint Q_{\mathcal{Z}}(\mathcal{Z}) Q_\theta(\boldsymbol{\theta}) \log P(\mathcal{X}, \mathcal{Z}, \boldsymbol{\theta}|\eta) \, \mathrm{d}\mathcal{Z} \, \mathrm{d}\boldsymbol{\theta}$$

Remark 35. **(ARD for Unsupervised Learning)** The hyperparameter method to select a useful input regression. This is similar idea with variational Bayes method that can learn a latent dimension. Consider the factor analysis:

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\Lambda z}, \boldsymbol{\Psi})$$

where $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ with column-wise prior to be $\boldsymbol{\Lambda}_{:i} \sim \mathcal{N}(\boldsymbol{0}, \alpha_i^{-1}\boldsymbol{I})$. Then the VB free-energy is given by:

$$\mathcal{F}\Big(Q_{\mathcal{Z}}(\mathcal{Z}), Q_\Lambda(\boldsymbol{\Lambda}), \boldsymbol{\Psi}, \boldsymbol{\alpha}\Big) = \Big\langle \log P(\mathcal{X}, \mathcal{Z}|\boldsymbol{\Lambda}, \boldsymbol{\Psi}) + \log P(\boldsymbol{\Lambda}|\boldsymbol{\alpha}) + \log P(\boldsymbol{\Psi}) \Big\rangle_{Q_{\mathcal{Z}}(\mathcal{Z})Q_\Lambda(\boldsymbol{\Lambda})} + \text{const.}$$

We require the hyperparameter optimization require us $\boldsymbol{\alpha} \leftarrow \arg\max \langle \log P(\boldsymbol{\Lambda}|\boldsymbol{\alpha}) \rangle_{Q_\Lambda}$. Now we obtain the following optimization:

- Now, Q_Λ is Gaussian with the same form as linear regression but with expected moment of $\boldsymbol{z}$ instead of the input.
- The optimization with respected to the distribution $\boldsymbol{\Psi}$ and $\boldsymbol{\alpha}$ will cause some α_i to diverge as in regression ARD.
- Same as selecting relevant latent dimension, effectively learning the dimension of latent variable.

Remark 36. **(Problem with GP + Solution)** Given the GP prediction, we have:

$$y'|\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{x} \sim \mathcal{GP}(\boldsymbol{K}_{xX}(\boldsymbol{K}_{XX} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{Y},\ \boldsymbol{K}_{xx} - \boldsymbol{K}_{xX}(\boldsymbol{K}_{XX} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{K}_{Xx} + \sigma^2)$$

The evidence (for learning the kernel hyperparameter) to be:

$$\log P(\boldsymbol{Y}|\boldsymbol{X}) = -\frac{1}{2}\log\left|2\pi(\boldsymbol{K}_{XX} + \sigma^2\boldsymbol{I})\right|^{-1/2} \exp\left(-\frac{1}{2}\boldsymbol{Y}^T(\boldsymbol{K}_{XX} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{Y}\right)$$

Computing this require inverting $N \times N$ matrix with has the time complexity of $\mathcal{O}(K^3)$. We consider the smaller set of possible fictious measurement $\boldsymbol{U}$ at input $\boldsymbol{Z}$ such that:

$$P(y'|\boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{x}') \approx P(y'|\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{x}')$$

Write $\boldsymbol{F}$ for the smooth GP function that underlie $\boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{F}, \sigma^2 I)$. Introduce the measurement $\boldsymbol{U}$ at input $\boldsymbol{Z}$, as the likelihood can be written as:

$$\begin{aligned} P(\boldsymbol{Y}|\boldsymbol{X}) &= \iint P(\boldsymbol{Y}, \boldsymbol{F}, \boldsymbol{U}|\boldsymbol{X}, \boldsymbol{Z})\ \mathrm{d}\boldsymbol{F}\ \mathrm{d}\boldsymbol{U} \\ &= \iint P(\boldsymbol{Y}|\boldsymbol{F})P(\boldsymbol{F}|\boldsymbol{U}, \boldsymbol{X}, \boldsymbol{Z})P(\boldsymbol{U}|\boldsymbol{Z})\ \mathrm{d}\boldsymbol{F}\ \mathrm{d}\boldsymbol{U} \end{aligned}$$

Remark 37. **(Free-Energy)** As we have $\boldsymbol{F}$ and $\boldsymbol{U}$ to be the latent variable, we introduce the variational distribution $q(\boldsymbol{F}, \boldsymbol{U})$ to be:

$$\mathcal{F}(q(\boldsymbol{F}, \boldsymbol{U}), \theta) = \left\langle \log \frac{P(\boldsymbol{Y}|\boldsymbol{F})P(\boldsymbol{F}|\boldsymbol{U}, \boldsymbol{X}, \boldsymbol{Z})P(\boldsymbol{U}|\boldsymbol{Z})}{q(\boldsymbol{F}, \boldsymbol{U})} \right\rangle_{q(\boldsymbol{F}, \boldsymbol{U})}$$

Consider the variational distribution of the latent to be $q(\boldsymbol{F}, \boldsymbol{U}) = P(\boldsymbol{F}|\boldsymbol{U}, \boldsymbol{X}, \boldsymbol{Z})q(\boldsymbol{U})$. We fix $\boldsymbol{F}|\boldsymbol{U}$ with reference to $\boldsymbol{U}$ to make the information about $\boldsymbol{Y}$ is compressed to $q(\boldsymbol{U})$, which we have:

$$\begin{aligned} \mathcal{F}(q(\boldsymbol{F}, \boldsymbol{U}), \theta, \boldsymbol{Z}) &= \left\langle \log \frac{P(\boldsymbol{Y}|\boldsymbol{F})P(\boldsymbol{F}|\boldsymbol{U}, \boldsymbol{X}, \boldsymbol{Z})P(\boldsymbol{U}|\boldsymbol{Z})}{P(\boldsymbol{F}|\boldsymbol{U}, \boldsymbol{X}, \boldsymbol{Z})q(\boldsymbol{U})} \right\rangle_{P(\boldsymbol{F}|\boldsymbol{U}, \boldsymbol{X}, \boldsymbol{Z})q(\boldsymbol{U})} \\ &= \left\langle \log \frac{P(\boldsymbol{Y}|\boldsymbol{F})P(\boldsymbol{U}|\boldsymbol{Z})}{q(\boldsymbol{U})} \right\rangle_{P(\boldsymbol{F}|\boldsymbol{U}, \boldsymbol{X}, \boldsymbol{Z})q(\boldsymbol{U})} \\ &= \Big\langle \langle \log P(\boldsymbol{Y}|\boldsymbol{F}) \rangle_{P(\boldsymbol{F}|\boldsymbol{U}, \boldsymbol{X}, \boldsymbol{Z})} + \log P(\boldsymbol{U}|\boldsymbol{Z}) - q(\boldsymbol{U}) \Big\rangle_{q(\boldsymbol{U})} \end{aligned}$$

Let's consider the inner expectation, which we can use the Gaussian process results (without noise on $\boldsymbol{F}$):

$$\begin{aligned}\langle \log P(\boldsymbol{Y}|\boldsymbol{F})\rangle_{P(\boldsymbol{F}|\boldsymbol{U},\boldsymbol{X},\boldsymbol{Z})} &= \left\langle -\frac{1}{2}\log\left|2\pi\sigma^2\boldsymbol{I}\right| - \frac{1}{2\sigma^2}\operatorname{Tr}\left[(\boldsymbol{Y}-\boldsymbol{F})(\boldsymbol{Y}-\boldsymbol{F})^T\right]\right\rangle_{P(\boldsymbol{F}|\boldsymbol{Y})} \\ &= -\frac{1}{2}\log\left|2\pi\sigma^2\boldsymbol{I}\right| - \frac{1}{2\sigma^2}\operatorname{Tr}\left[\left(\boldsymbol{Y}-\langle\boldsymbol{F}\rangle_{P(\boldsymbol{F}|\boldsymbol{Y})}\right)\left(\boldsymbol{Y}-\langle\boldsymbol{F}\rangle_{P(\boldsymbol{F}|\boldsymbol{Y})}\right)^T\right] - \frac{1}{2\sigma^2}\operatorname{Tr}\left[\boldsymbol{\Sigma}_{F|U}\right] \\ &= \log\mathcal{N}(\boldsymbol{Y}|\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{U},\sigma^2\boldsymbol{I}) - \frac{1}{2\sigma^2}\operatorname{Tr}[\boldsymbol{K}_{XX}-\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{K}_{ZX}]\end{aligned}$$

This gives us the following free energy function:

$$\mathcal{F}(q(\boldsymbol{U}),\boldsymbol{\theta},\boldsymbol{Z}) = \left\langle \log\frac{\mathcal{N}(\boldsymbol{Y}|\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{U},\ \sigma^2\boldsymbol{I})P(\boldsymbol{U}|\boldsymbol{Z})}{q(\boldsymbol{U})}\right\rangle_{q(\boldsymbol{U})} - \frac{1}{2\sigma^2}\operatorname{Tr}[\boldsymbol{K}_{XX}-\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{K}_{ZX}]$$

Now, we can see that the expectation is free energy of PPCA-like model with the latent to be $\boldsymbol{U}\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{K}_{ZZ})$, with the following linear Gaussian model $p(\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{U},\sigma^2\boldsymbol{I})$, thusm we have the following free energy:

$$\begin{aligned}\mathcal{F}(q^*(\boldsymbol{U}),\boldsymbol{\theta},\boldsymbol{Z}) &= \log\mathcal{N}(\boldsymbol{Y}|\boldsymbol{0},\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{K}_{ZZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{K}_{ZX}+\sigma^2\boldsymbol{I}) - \frac{1}{2\sigma^2}\operatorname{Tr}[\boldsymbol{K}_{XX}-\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{K}_{ZX}] \\ &= \log\mathcal{N}(\boldsymbol{Y}|\boldsymbol{0},\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{K}_{ZX}+\sigma^2\boldsymbol{I}) - \frac{1}{2\sigma^2}\operatorname{Tr}[\boldsymbol{K}_{XX}-\boldsymbol{K}_{XZ}\boldsymbol{K}_{ZZ}^{-1}\boldsymbol{K}_{ZX}]\end{aligned}$$

We can now optimize the free energy numerically with respect to $\boldsymbol{\theta}$ and $\boldsymbol{Z}$ to adjust GP prior and quality of variational approximation.

Remark 38. If $\boldsymbol{X}$ is unobserved, then assume that $q(\boldsymbol{X},\boldsymbol{F},\boldsymbol{U}) = q(\boldsymbol{X})P(\boldsymbol{F},\boldsymbol{X},\boldsymbol{U})q(\boldsymbol{U})$ then we have the free energy to be:

$$\mathcal{F} = \langle \log P(\boldsymbol{Y},\boldsymbol{F},\boldsymbol{U}|\boldsymbol{X}) + \log P(\boldsymbol{X})\rangle_{q(\boldsymbol{U})q(\boldsymbol{X})}$$

which can be simplifed to tractable components in similar way as above.

5 Expectation Propagation

5.1 Introduction

Remark 39. **(Non-Linear State Space)** We are given the following non-linear space with the following transition and emission probability as we have:

$$\boldsymbol{z}_{t+1} = f(\boldsymbol{z}_t,\boldsymbol{u}_t) + \boldsymbol{w}_t \qquad \boldsymbol{x}_t = g(\boldsymbol{z}_t,\boldsymbol{u}_t) + \boldsymbol{v}_t$$

where $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are usually Gaussian, as we can linearlized the non-linear function around $\hat{\boldsymbol{z}}_t^t$:

$$\boldsymbol{z}_{t+1} \approx \underbrace{f(\hat{\boldsymbol{z}}_t^t,\boldsymbol{u}_t)}_{\tilde{\boldsymbol{B}}_t\boldsymbol{u}_t} + \underbrace{\left.\frac{\partial f}{\partial \boldsymbol{z}_t}\right|_{\hat{\boldsymbol{z}}_t^t}}_{\tilde{\boldsymbol{A}}_t}(\boldsymbol{z}_t-\hat{\boldsymbol{z}}_t^t) + \boldsymbol{w}_t \qquad \boldsymbol{x}_t \approx \underbrace{f(\hat{\boldsymbol{z}}_t^{t-1},\boldsymbol{u}_t)}_{\tilde{\boldsymbol{D}}_t\boldsymbol{u}_t} + \underbrace{\left.\frac{\partial f}{\partial \boldsymbol{z}_t}\right|_{\hat{\boldsymbol{z}}_t^{t-1}}}_{\tilde{\boldsymbol{C}}_t}(\boldsymbol{z}_t-\hat{\boldsymbol{z}}_t^{t-1}) + \boldsymbol{w}_t$$

We can run the kalman filter on non-stationary linearized system $(\tilde{\boldsymbol{A}}_t,\tilde{\boldsymbol{B}}_t,\tilde{\boldsymbol{C}}_t,\tilde{\boldsymbol{D}}_t)$:

- Adaptive approximate non-Gaussian messaage by Gaussian.
- Local linearization depends on central point of distribution, mening that approximation degrads with more stable uncertainty.
- This might work in the system that is close to linear.

Remark 40. **(Other Message Approximation)** We have the following message on latent chain:

$$P(\boldsymbol{z}_t|\boldsymbol{x}_{1:t}) = \frac{1}{Z}P(\boldsymbol{x}_t|\boldsymbol{z}_t)\int P(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})P(\boldsymbol{z}_{t-1}|\boldsymbol{x}_{1:t-1})\ \mathrm{d}\boldsymbol{z}_{t-1}$$
$$\tilde{P}(\boldsymbol{z}_{t-1}|\boldsymbol{x}_{1:t-1}) \approx \frac{1}{Z}P(\boldsymbol{x}_t|\boldsymbol{z}_t)\int \underbrace{P(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})}_{\mathcal{N}(f(\boldsymbol{z}_{t-1}),\boldsymbol{Q})}\ \underbrace{\tilde{P}(\boldsymbol{z}_{t-1}|\boldsymbol{x}_{1:t-1})}_{\mathcal{N}(\hat{\boldsymbol{z}}_{t-1},\hat{\boldsymbol{V}}_{t-1})}\ \mathrm{d}\boldsymbol{z}_{t-1}$$

There are several, way to approximate the integration: linearization at the peak (EKF) is only one approach. We can use the Laplace filter use mode and curvature of integrand. Finally, we can use the sigma-point.

Remark 41. **(Why not KL?)** We can consider the parametric variational as we have

$$\mathrm{KL}\left[\mathcal{N}(\hat{\boldsymbol{z}}_t,\hat{\boldsymbol{V}})\middle\|\int P(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})P(\boldsymbol{z}_{t-1}|\boldsymbol{x}_{1:t-1})\ \mathrm{d}\boldsymbol{z}_{t-1}\right]$$

This may be hard to find the closed form solution to this KL divergence and we might have to find the value using Monte-Carlo sampling.

5.2 Expectation Propagation

Proposition 5.1. *Given the $p(\boldsymbol{x})$, we let q to be exponential family, where we have:*

$$Q(\boldsymbol{x}) = \frac{\exp\left(\boldsymbol{T}(\boldsymbol{x})^T\boldsymbol{\theta}\right)}{Z(\boldsymbol{\theta})}$$

Now, consider the following minimization of: $Q^* = \arg\min_Q \mathrm{KL}\left[P(\boldsymbol{x})\middle\|Q(\boldsymbol{x})\right]$ is solved when:

$$\langle \boldsymbol{T}(\boldsymbol{x})\rangle_{Q^*} = \langle \boldsymbol{T}(\boldsymbol{x})\rangle_P$$

Or matches the sufficient statistics.

Proof. Consider the KL-divergence to be:

$$\begin{aligned}
\arg\min_Q\ \mathrm{KL}\left[P(\boldsymbol{x})\middle\|Q(\boldsymbol{x})\right] &= \arg\min_{\boldsymbol{\theta}}\ \left[P(\boldsymbol{x})\middle\|\frac{\exp\left(\boldsymbol{T}(\boldsymbol{x})^T\boldsymbol{\theta}\right)}{Z(\boldsymbol{\theta})}\right] \\
&= \arg\min_{\boldsymbol{\theta}}\ -\int P(\boldsymbol{x})\log\left(\frac{\exp\left(\boldsymbol{T}(\boldsymbol{x})^T\boldsymbol{\theta}\right)}{Z(\boldsymbol{\theta})}\right)\ \mathrm{d}\boldsymbol{x} \\
&= \arg\min_{\boldsymbol{\theta}}\ \log(Z(\boldsymbol{\theta})) - \int P(\boldsymbol{x})\left[\boldsymbol{T}(\boldsymbol{x})^T\boldsymbol{\theta}\right]\ \mathrm{d}\boldsymbol{x}
\end{aligned}$$

The second equality comes from the fact that $\int P(\boldsymbol{x})\log P(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x}$ is constant. Now, we have the following derivative with respected to $\boldsymbol{\theta}$:

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{\theta}}\left(\log(Z(\boldsymbol{\theta})) - \int P(\boldsymbol{x})\left[\boldsymbol{T}(\boldsymbol{x})^T\boldsymbol{\theta}\right]\ \mathrm{d}\boldsymbol{x}\right) &= \frac{1}{Z(\boldsymbol{\theta})}\frac{\partial}{\partial\boldsymbol{\theta}}\int \exp(\boldsymbol{T}(\boldsymbol{x})^T\boldsymbol{\theta})\ \mathrm{d}\boldsymbol{x} - \int p(\boldsymbol{x})T(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} \\
&= \frac{1}{Z(\boldsymbol{\theta})}\int \exp(\boldsymbol{T}(\boldsymbol{x})^T\boldsymbol{\theta})T(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} - \langle T(\boldsymbol{x})\rangle_P \\
&= \langle T(\boldsymbol{x})\rangle_P - \langle T(\boldsymbol{x})\rangle_Q
\end{aligned}$$

Setting this to zero as we have the solution. □

Remark 42. **(KL-Divergence Solution)** It is better for use to perform the following KL minimization:

$$\mathrm{KL}\left[\int P(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})P(\boldsymbol{z}_{t-1}|\boldsymbol{x}_{1:t-1})\ \mathrm{d}\boldsymbol{z}_{t-1}\middle\|\mathcal{N}(\hat{\boldsymbol{z}}_t,\hat{\boldsymbol{V}})\right]$$

As we can consider the expected sufficient statistics of the real distribution, instead of finding the exact version of it. However, to calculate the expected sufficient statistics $\langle \boldsymbol{T}(\boldsymbol{x})\rangle_P$, which may be analytically tractable, however for the high-dimensional integral (we can use various algorithms to calculate the expectation).

Remark 43. **(Another Problem of KL-Divergence)** Let's consider the KL-divergence of the factored model, as we have:

$$\begin{aligned}
\arg\min_{q_i}\ \mathrm{KL}\left[P(\mathcal{Z}|\mathcal{X})\middle\|\prod_j q_j(\mathcal{Z}_j|\mathcal{X})\right] &= \arg\min_{q_i}\ -\int P(\mathcal{Z}|\mathcal{X})\log\prod_j q_j(\mathcal{Z}_j|\mathcal{X})\ \mathrm{d}\mathcal{Z}\\
&= \arg\min_{q_i}\ -\sum_j\int P(\mathcal{Z}|\mathcal{X})\log q_j(\mathcal{Z}_j|\mathcal{X})\ \mathrm{d}\mathcal{Z}\\
&= \arg\min_{q_i}\ -\int P(\mathcal{Z}_i|\mathcal{X})\log q_i(\mathcal{Z}_i|\mathcal{X})\ \mathrm{d}\mathcal{Z}_i\\
&= P(\mathcal{Z}_i|\mathcal{X})
\end{aligned}$$

We will have to know the marginal, which is intractable.

Remark 44. **(EP Motivation)** The posterior distribution in a graphical model is a product of factors:

$$P(\mathcal{Z}|\mathcal{X}) = \frac{P(\mathcal{Z},\mathcal{X})}{P(\mathcal{X})} = \frac{1}{Z}\prod_i P(\mathcal{Z}_i|\,\mathrm{Pa}(\mathcal{Z}_i)) \propto \prod_i f_i(\mathcal{Z}_i)$$

where Z_i isn't necessary disjoint and we call f_i sites, we consider the same factorization, but with approximate site as we have:

$$\min_{\tilde{f}_i}\mathrm{KL}\left[f_i(\mathcal{Z}_i)\prod_{j\neq i}\tilde{f}_j(\mathcal{Z}_j)\middle\|\tilde{f}_i(\mathcal{Z}_i)\prod_{j\neq i}\tilde{f}_j(\mathcal{Z}_j)\right] \iff \min_f \mathrm{KL}\left[f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})\middle\|f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})\right]$$

where we have $q_{\neg i}(\mathcal{Z}) = \prod_{j\neq i}\tilde{f}_j(\mathcal{Z}_j)$. This leads to 2 ideas about expectation propagation as:

- *Expectation*: Approximation of factors, where we perform a projection to exponential family, thus requiring a sufficient statistics.
- *Propagation*: Local divergence minimization leads to message passing approach (hence the number of propagation).

Proposition 5.2. ***(Simplier Update)*** *If we consider the context factor, as* $q_{\neg i}(\mathcal{Z}) = q_{\neg i}(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_{\neg i}|\mathcal{Z}_i)$ *and we have* $\mathcal{Z}_{\neg i} = \mathcal{Z}\backslash\mathcal{Z}_i$ *we can show that:*

$$\min_f \mathrm{KL}\left[f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})\middle\|f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})\right] \equiv \min_f \mathrm{KL}\left[f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\middle\|f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\right]$$

Please see that the differences between those 2 *between red and blue. We call* $q_{\neg i}(\mathcal{Z}_i)$

Proof. Consider the following equalities:

$$\begin{aligned}
\min_f \mathrm{KL}\left[f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})\middle\|f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})\right] &= \max_f \int f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})\log\left[f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})\right]\,\mathrm{d}\mathcal{Z} \\
&= \max_f \int f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_{\neg i}|\mathcal{Z}_i)\log\left[f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_{\neg i}|\mathcal{Z}_i)\right]\,\mathrm{d}\mathcal{Z}_i\,\mathrm{d}\mathcal{Z}_{\neg i} \\
&= \max_f \int f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_{\neg i}|\mathcal{Z}_i)\left[\log f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\right]\,\mathrm{d}\mathcal{Z}_i\,\mathrm{d}\mathcal{Z}_{\neg i} \\
&= \max_f \int f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\left[\log f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\right]\,\mathrm{d}\mathcal{Z}_i\int q_{\neg i}(\mathcal{Z}_{\neg i}|\mathcal{Z}_i)\,\mathrm{d}\mathcal{Z}_{\neg i} \\
&= \min_f \mathrm{KL}\left[f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\middle\|f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\right]
\end{aligned}$$

And so the equality is proven. □

Definition 5.1. (Expectation Propagation) We consider the following algorithms for expectation propagation, pseudocode:

Algorithm 1 Expectation Propagation Algorithm

1: **Input**: $f_1(\mathcal{Z}_1),\ldots,f_N(\mathcal{Z}_N)$
2: **Initilize**: $\tilde{f}_i(\mathcal{Z}_i)=1$ for $i\in[n]$ and we have $q(\mathcal{Z})\propto\prod_i\tilde{f}_i(\mathcal{Z}_i)$
3: **while** convergence **do**
4: **for** $i=1,2,\cdots,N$ **do**
5: *Delete*: We do the update to get the cavity

$$q_{\neg i}(\mathcal{Z})\leftarrow\frac{q(\mathcal{Z})}{\tilde{f}_i(\mathcal{Z}_i)}=\prod_{j\neq i}\tilde{f}_j(\mathcal{Z}_j)$$

6: *Project*: Performing KL-divergence minimization

$$\tilde{f}_i^{\text{new}}(\mathcal{Z})\leftarrow\arg\min_f\ \mathrm{KL}\left[f_i(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\middle\|f(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z}_i)\right]$$

7: *Include*: $q(\mathcal{Z})\leftarrow\tilde{f}_i^{\text{new}}(\mathcal{Z}_i)q_{\neg i}(\mathcal{Z})$
8: **end for**
9: **end while**

Remark 45. **(Calculate the Cavity Distribution)** The cavity distribution can be broken down into product of terms for each neighboring cliques:

$$q_{\neg i}(\mathcal{Z}_i)=\prod_{j\in\mathrm{ne}(i)}M_{j\to i}(\mathcal{Z}_i\cap\mathcal{Z}_j)$$

The i-th site has been approximated as the message can be passed onto neighboring cliques by normalizing the shared variable. This is belief propagation. Furthermore, the message updates can be scheduled in any order. However, there is no guarantee of convergence.

Remark 46. **(Normalizer)** As long as approximating class is tractable, normalizer can be computed as needed. Consider the approximation class to be:

$$\tilde{f}_i(\mathcal{Z}_i)\propto\exp\left(\boldsymbol{T}(\mathcal{Z})^T\boldsymbol{\theta}_i-\Phi(\boldsymbol{\theta}_i)\right)$$

This is the same as setting every entries $\boldsymbol{\theta}_i$ to 0 except for the one that is in $\mathcal{Z}_i$ (while finding the sufficient statistics for all latents). This will give us the following probability:

$$q(\mathcal{Z})\propto\prod_i\tilde{f}_i\propto\exp\left(\boldsymbol{T}(\mathcal{Z})^T\sum_i\boldsymbol{\theta}_i-\sum_i\Phi(\boldsymbol{\theta}_i)\right)$$

We can re-normalizied it as we have $q(\mathcal{Z}) = \exp\left(\boldsymbol{T}(\mathcal{Z})^T \sum_i \boldsymbol{\theta}_i - \Phi\left(\sum_i \boldsymbol{\theta}_i\right)\right)$

Remark 47. **(Computing Likelihood)** Consider the unnormalized exponential family, where we have the approximating site to be:

$$\tilde{f}_i = \tilde{C}_i \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_i\right)$$

We consider the following values

- $\boldsymbol{\theta} = \sum_i \boldsymbol{\theta}_i$, which is the natural parameter of $q(\mathcal{Z})$
- $\boldsymbol{\theta}_{\neg i} = \sum_{j \neq i} \boldsymbol{\theta}_i$ as the natural parameter of $q_{\neg i}(\mathcal{Z})$
- Exponential family (tractable) log-normalizer $\boldsymbol{\Phi}(\boldsymbol{\theta})$ of $P(\mathcal{Z})$ to be:

$$\boldsymbol{\Phi}(\boldsymbol{\theta}) = \log \int \exp(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}) \, \mathrm{d}\mathcal{Z}$$

Now, we are interested to find the actual approximation of the normalizer i.e:

$$\log \int \prod_{i=1}^N f_i(\mathcal{Z}_i)$$

To do this, we minimize the unnormalized KL divergence (where we also care about the normalizer, see the additional term):

$$\mathrm{KL}_{\mathrm{un}}[p \| q] = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x} + \int \Big(q(\boldsymbol{x}) - p(\boldsymbol{x})\Big) \, \mathrm{d}\boldsymbol{x}$$

Proposition 5.3. ***(Log Likelihood Approximation)*** *The approximation of the log-likelihood is given as:*

$$\log \int \prod_{i=1}^N f_i(\mathcal{Z}_i) \approx \log \int \prod_{i=1}^N \tilde{f}_i(\mathcal{Z}_i) = (1-N)\Phi(\boldsymbol{\theta}) + \sum_{i=1}^N \Phi_i(\boldsymbol{\theta}_{\neg i})$$

where $\boldsymbol{\Phi}_i$ is the log-normalizer of the distribution $\hat{P}_i(\mathcal{Z}) \propto f(\mathcal{Z}) \exp(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta})$

Proof. To follows the unnormalized KL divergence, the first terms simply follows the expected sufficient statistics match. In this case, we have (thinking of it as having KL-divergence equal to zero):

$$\begin{aligned}
&\min_{\tilde{C}_i} \mathrm{KL}_{\min}\left[\tilde{C}_i \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_i\right) \prod_{\neg i} \tilde{C}_j \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_j\right) \middle\| f_i(\mathcal{Z}_i) \prod_{\neg i} \tilde{C}_j \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_j\right)\right] \\
\iff &\int \tilde{C}_i \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_i\right) \prod_{\neg i} \tilde{C}_j \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_j\right) \mathrm{d}\mathcal{Z} = \int f_i(\mathcal{Z}_i) \prod_{\neg i} \tilde{C}_j \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_j\right) \mathrm{d}\mathcal{Z} \\
\iff &\int \tilde{C}_i \prod_i \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_j\right) \mathrm{d}\mathcal{Z} = \int f_i(\mathcal{Z}_i) \prod_{\neg i} \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_j\right) \mathrm{d}\mathcal{Z} \\
\iff &\tilde{C}_i \exp(\boldsymbol{\Phi}(\boldsymbol{\theta})) = \exp(\boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i})) \\
\iff &\tilde{C}_i = \exp\Big(\boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}) - \boldsymbol{\Phi}(\boldsymbol{\theta})\Big)
\end{aligned}$$

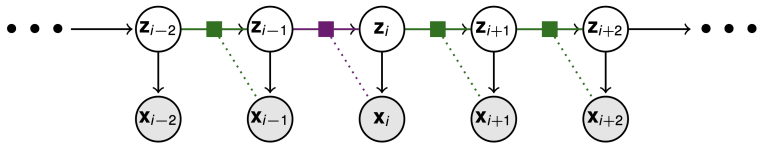
and so, we have the following log-likelihood to be:

$$\begin{aligned}
\log \int \prod_{i=1}^N \tilde{f}_i(\mathcal{Z}_i) \, \mathrm{d}\mathcal{Z} &= \log \int \prod_{i=1}^N \tilde{C}_i \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_i\right) \mathrm{d}\mathcal{Z} \\
&= \log \prod_{i=1}^N \tilde{C}_i \int \exp\left(\boldsymbol{T}(\mathcal{Z})^T \left[\sum_{i=1}^N \boldsymbol{\theta}_i\right]\right) \mathrm{d}\mathcal{Z} \\
&= \boldsymbol{\Phi}(\boldsymbol{\theta}) + \sum_{i=1}^N \log \tilde{C}_i = (1-N)\Phi(\boldsymbol{\theta}) + \sum_{i=1}^N \Phi_i(\boldsymbol{\theta}_{\neg i})
\end{aligned}$$

□

5.3 Examples

Remark 48. **(Example: Non-Linear State Space Model)** Consider the graphical model:



where we have the following factors:

$$P(\boldsymbol{z}_i|\boldsymbol{z}_{i-1}) = \phi_i(\boldsymbol{z}_i, \boldsymbol{z}_{i-1}) = \exp\left(-\frac{\|\boldsymbol{z}_i - h_s(\boldsymbol{z}_i)\|^2}{2\sigma^2}\right)$$
$$P(\boldsymbol{x}_i|\boldsymbol{z}_i) = \psi_i(\boldsymbol{z}_i) = \exp\left(-\frac{\|\boldsymbol{x}_i - h_o(\boldsymbol{z}_i)\|^2}{2\sigma^2}\right)$$

We can see that $f_i(\boldsymbol{z}_i, \boldsymbol{z}_{i-1}) = \phi_i(\boldsymbol{z}_i, \boldsymbol{z}_{i-1})\psi_i(\boldsymbol{z}_i)$ as there are non-linear, the inference isn't generally tractable. For the EP, we will consider the approximate $\tilde{f}(\boldsymbol{z}_i, \boldsymbol{z}_{i-1})$ to be Gaussian as we have. Consider the cavity distribution to be:

$$\begin{aligned} q_{\neg i}(\boldsymbol{z}_i, \boldsymbol{z}_{i-1}) &= \int_{\boldsymbol{z}_1,\ldots,\boldsymbol{z}_{i-2},\boldsymbol{z}_{i+1},\ldots,\boldsymbol{z}_n} \prod_{i' \neq i} \tilde{f}_{i'}(\boldsymbol{z}_{i'}, \boldsymbol{z}_{i'-1}) \\ &= \underbrace{\int_{\boldsymbol{z}_1,\ldots,\boldsymbol{z}_{i-2}} \prod_{i'<i} \tilde{f}_{i'}(\boldsymbol{z}_{i'}, \boldsymbol{z}_{i'-1})}_{\alpha_{i-1}(\boldsymbol{z}_{i-1})} \underbrace{\int_{\boldsymbol{z}_{i+1},\ldots,\boldsymbol{z}_n} \prod_{i'>i} \tilde{f}_{i'}(\boldsymbol{z}_{i'}, \boldsymbol{z}_{i'-1})}_{\beta_i(\boldsymbol{z}_i)} \end{aligned}$$

Please note that α, β is being Gaussian (by default), and so we have the following update rule:

$$\tilde{f}_i(\boldsymbol{z}_i, \boldsymbol{z}_{i-1}) = \arg\min_{f \in \mathcal{N}} \ \mathrm{KL}\left[\phi_i(\boldsymbol{z}_i, \boldsymbol{z}_{i-1})\psi_i(\boldsymbol{z}_i)\alpha_{i-1}(\boldsymbol{z}_{i-1})\beta_i(\boldsymbol{z}_i) \middle\| f(\boldsymbol{z}_i, \boldsymbol{z}_{t-1})\alpha_{i-1}(\boldsymbol{z}_{i-1})\beta_i(\boldsymbol{z}_i)\right]$$

We can consider the following optimization instead as we have:

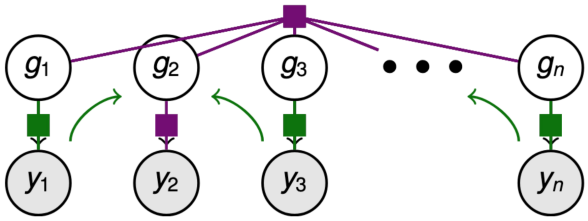
$$\begin{aligned} \tilde{P}(\boldsymbol{z}_{i-1}, \boldsymbol{z}_i) &= \arg\min_{P \in \mathcal{N}} \ \mathrm{KL}\left[\hat{P}(\boldsymbol{z}_{i-1}, \boldsymbol{z}_i) \middle\| P(\boldsymbol{z}_{i-1}, \boldsymbol{z}_i)\right] \\ \implies \tilde{f}_i(\boldsymbol{z}_i, \boldsymbol{z}_{i-1}) &= \frac{\tilde{P}(\boldsymbol{z}_{i-1}, \boldsymbol{z}_i)}{\alpha_{i-1}(\boldsymbol{z}_{i-1})\beta_i(\boldsymbol{z}_i)} \end{aligned}$$

Now, we consider each values $\alpha_i(\boldsymbol{z}_i)$ and $\beta_{i-1}(\boldsymbol{z}_{t-1})$ as the propagation step:

$$\alpha_i(\boldsymbol{z}_i) = \int_{\boldsymbol{z}_1,\ldots,\boldsymbol{z}_{i-1}} \prod_{i'<i+1} \tilde{f}_{i'}(\boldsymbol{z}_{i'}, \boldsymbol{z}_{i'-1}) = \int_{\boldsymbol{z}_i} \alpha_{i-1}(\boldsymbol{z}_{i-1})\tilde{f}_i(\boldsymbol{z}_i, \boldsymbol{z}_{i-1}) = \frac{1}{\beta_i(\boldsymbol{z}_i)} \int_{\boldsymbol{z}_{i-1}} \tilde{P}(\boldsymbol{z}_{i-1}, \boldsymbol{z}_i)\ \mathrm{d}\boldsymbol{z}_{i-1}$$
$$\beta_{i-1}(\boldsymbol{z}_{i-1}) = \int_{\boldsymbol{z}_i,\ldots,\boldsymbol{z}_n} \prod_{i'<i+1} \tilde{f}_{i'}(\boldsymbol{z}_{i'}, \boldsymbol{z}_{i'-1}) = \int_{\boldsymbol{z}_i} \beta_i(\boldsymbol{z}_i)\tilde{f}_i(\boldsymbol{z}_i, \boldsymbol{z}_{i-1}) = \frac{1}{\alpha_{i-1}(\boldsymbol{z}_{i-1})} \int_{\boldsymbol{z}_i} \tilde{P}(\boldsymbol{z}_{i-1}, \boldsymbol{z}_i)\ \mathrm{d}\boldsymbol{z}_i$$

The last equality comes from the another definition of $\tilde{f}_i(\boldsymbol{z}_{i-1}, \boldsymbol{z}_i)$. The update of both α_i and β_i are marginalization of $\tilde{P}(\boldsymbol{z}_{i-1}, \boldsymbol{z}_i)$, which is also a Gaussian.

Remark 49. **(EP For GP Classification)** We can write the GP joint on g_i and y_i as a factor graph:



Consider the following factorization:

$$P(g_1, \ldots, g_n, y_1, \ldots, y_n) = \underbrace{\mathcal{N}(g_1, \ldots, g_n | 0, K)}_{f_0(\mathcal{G})} \prod_i \underbrace{P(y_i|g_i)}_{f_i(g_i)}$$

Some factorization applied to non-Gaussian $P(y_i = 1|g_i) = 1/(1+\exp(-g_i))$. EP approximate non-Gaussian $f_i(g_i)$ by Gaussian to be $\tilde{f}_i(g_i) = \mathcal{N}(\tilde{\mu}_i, \tilde{\psi}_i^2)$. Given $q_{\neg i}(g_i)$ can be concluded by GP marginalization:

$$q_{\neg i}(g_i) = \mathcal{N}\Big(\boldsymbol{\Sigma}_{i,\neg i}\boldsymbol{\Sigma}_{\neg i,\neg i}^{-1}\tilde{\boldsymbol{\mu}}_{\neg i},\ K_{i,i} - \boldsymbol{\Sigma}_{i,\neg i}\boldsymbol{\Sigma}_{\neg i,\neg i}^{-1}\boldsymbol{\Sigma}_{\neg i,i}\Big)$$

We will consider $\boldsymbol{\Sigma} = \boldsymbol{K} + \text{diag}(\tilde{\psi}_1^2, \ldots, \tilde{\psi}_n^2)$. The update on the Gaussian is based on sufficient statistics matching where $P(g) = f_i(g)q_\neg(g)$, as now we have the follow:

$$\tilde{f}_{\text{new}}(g_i) = \mathcal{N}\Big(\underbrace{\int q_{\neg i}(g) f_i(g) g \ \mathrm{d}g}_{\tilde{\mu}_i^{\text{new}}},\ \int q_{\neg i}(g) f_i(g) g^2 \ \mathrm{d}g - (\tilde{\mu}_i^{\text{new}})^2\Big) \Big/ q_{\neg i}(g_i)$$

Once approximation site potential have stabilize, we use them to make a prediction from $\boldsymbol{x}'$. There are some observations that we made:

- Introduce the prediction point will changes $\boldsymbol{K}$, but it will not affect the marginal.
- The unobserved output factor proved no information about g' (the output of $\boldsymbol{x}'$) and so we don't have to approximate potential $\tilde{f}_i$
- And so, we have the following prediction:

  $$P(y'|\boldsymbol{x}', \mathcal{D}) = \int P(y'|\boldsymbol{g}') \mathcal{N}\left(\boldsymbol{g}' \middle| \boldsymbol{K}_{x'X}(\boldsymbol{K}_{XX} + \tilde{\boldsymbol{\Psi}})^{-1}\tilde{\boldsymbol{\mu}},\ \boldsymbol{K}_{x'x'}(\boldsymbol{K}_{XX} + \tilde{\boldsymbol{\Psi}})^{-1}\boldsymbol{K}_{Xx'}\right) \mathrm{d}\boldsymbol{g}'$$

  where $\tilde{\boldsymbol{\Psi}} = \text{diag}(\tilde{\psi}_1^2, \ldots, \tilde{\psi}_n^2)$.

5.4 Learning with EP

Remark 50. **(Learning and EP)** EP can yields the approximate inference posterior. To learn the hyper-parameter as we can use:

- Approximate Bayesian inference (like VB): Maybe need to construct a coherent normalizable exponential familty on both latent and parameter.
- Approximate EM: As we can maximize $\langle \log P(\mathcal{X}, \mathcal{Z}) \rangle_{q_{\text{EP}}(\mathcal{Z})}$. It is practical but not coherent cost function so no guarantee of convergence even if EP itself converges.
- Direct maximization of EP log-likelihood estimate:
  - Consistent although convergence guarantee still difficult.

– Seem hard to differentiate, but it is simpler than what it looks like.

Proposition 5.4. *We can show that the EP of the moment matching as we now have:*

$$\nabla_\eta \tilde{l} = \sum_{i=1}^{N} \langle \nabla_\eta \log f_i(\mathcal{Z}_i) \rangle_{\hat{P}_i}$$

where η is the model hyperparameter. It can be computed provided that EP converges.

Proof. Let's consider the derivative of $\log \tilde{C}_i$ as we have:

$$\nabla_\eta \log \tilde{C}_i = \nabla_\eta \boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}) - \nabla_\eta \boldsymbol{\Phi}(\boldsymbol{\theta}) = \nabla_\eta \boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}) - \boldsymbol{\mu}^T \nabla_\eta \boldsymbol{\theta}$$

Please recall the gradient of the log normalizer in the first equality i.e $\boldsymbol{\mu} = \langle \boldsymbol{T}(\mathcal{Z}) \rangle_{q(\mathcal{Z})}$. We will have to consider $\nabla_\eta \boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i})$ as we have (it depends on η via f_i and $\boldsymbol{\theta}_{\neg i}$):

$$\begin{aligned}
\nabla_\eta \boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}) &= \frac{1}{\exp(\boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}))} \left[\int \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_{\neg i}\right) \nabla_\eta f_i(\mathcal{Z}_i) \, \mathrm{d}\mathcal{Z} + \int f_i(\mathcal{Z}_i) \nabla_\eta \exp\left[\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_{\neg i}\right] \right] \\
&= \frac{1}{\exp(\boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}))} \int \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_{\neg i}\right) \nabla_\eta f_i(\mathcal{Z}_i) \, \mathrm{d}\mathcal{Z} + \int \frac{f_i(\mathcal{Z}_i) \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_{\neg i}\right)}{\exp(\boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}))} \boldsymbol{T}(\mathcal{Z})^T \nabla_\eta \boldsymbol{\theta}_{\neg i} \, \mathrm{d}\mathcal{Z} \\
&= \int \frac{f_i(\mathcal{Z}_i) \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_{\neg i}\right)}{\exp(\boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}))} \nabla_\eta \log f_i(\mathcal{Z}_i) \, \mathrm{d}\mathcal{Z} + \int \frac{f_i(\mathcal{Z}_i) \exp\left(\boldsymbol{T}(\mathcal{Z})^T \boldsymbol{\theta}_{\neg i}\right)}{\exp(\boldsymbol{\Phi}_i(\boldsymbol{\theta}_{\neg i}))} \boldsymbol{T}(\mathcal{Z})^T \nabla_\eta \boldsymbol{\theta}_{\neg i} \, \mathrm{d}\mathcal{Z} \\
&= \langle \nabla_\eta \log f_i(\mathcal{Z}_i) \rangle_{\hat{P}_i} + \langle \boldsymbol{T}(\mathcal{Z}) \rangle_{\hat{P}_i}^T \nabla_\eta \boldsymbol{\theta}_{\neg i} \\
&= \langle \nabla_\eta \log f_i(\mathcal{Z}_i) \rangle_{\hat{P}_i} + \boldsymbol{\mu}^T \nabla_\eta \boldsymbol{\theta}_{\neg i}
\end{aligned}$$

Consider the derivative with respected to the hyperparameter as we have:

$$\begin{aligned}
\nabla_\eta \tilde{l} &= \nabla_\eta \boldsymbol{\Phi}(\boldsymbol{\theta}) + \sum_{i=1}^{N} \nabla_\eta \log \tilde{C}_i \\
&= \boldsymbol{\mu}^T \nabla_\eta \boldsymbol{\theta} + \sum_{i=1}^{N} \Big(\langle \nabla_\eta \log f_i(\mathcal{Z}_i) \rangle_{\hat{P}_i} + \boldsymbol{\mu}^T \nabla_\eta \boldsymbol{\theta}_{\neg i} - \boldsymbol{\mu}^T \nabla_\eta \boldsymbol{\theta} \Big) \\
&= \boldsymbol{\mu}^T \nabla_\eta \left(\sum_{i=1}^{N} \boldsymbol{\theta}_i + \sum_{i=1}^{N} (\boldsymbol{\theta}_{\neg i} - \boldsymbol{\theta}) \right) + \sum_{i=1}^{N} \Big(\langle \nabla_\eta \log f_i(\mathcal{Z}_i) \rangle_{\hat{P}_i} \Big) \\
&= \boldsymbol{\mu}^T \nabla_\eta \left(\sum_{i=1}^{N} (\boldsymbol{\theta} - \boldsymbol{\theta}) \right) + \sum_{i=1}^{N} \langle \nabla_\eta \log f_i(\mathcal{Z}_i) \rangle_{\hat{P}_i} \\
&= \sum_{i=1}^{N} \langle \nabla_\eta \log f_i(\mathcal{Z}_i) \rangle_{\hat{P}_i}
\end{aligned}$$

Thus complete the proof □

Definition 5.2. (Alpha Divergence) This is a generalization of the KL-divergence to be:

$$D_\alpha[p \| q] = \frac{1}{\alpha(1-\alpha)} \int \alpha p(\boldsymbol{x}) + (1-\alpha) q(\boldsymbol{x}) - p(\boldsymbol{x})^\alpha q(\boldsymbol{x})^{1-\alpha} \, \mathrm{d}\boldsymbol{x}$$

Remark 51. We have the following values for α as we have:

- For $\alpha = -1$:

$$D_{-1}[p \| q] = \frac{1}{2} \int \frac{(p(\boldsymbol{x}) - q(\boldsymbol{x}))^2}{p(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x}$$

- For $\alpha = 1/2$:
$$D_{1/2}[p\|q] = 2\int \left(p(\boldsymbol{x})^{1/2} - q(\boldsymbol{x})^{1/2}\right)^2 \, \mathrm{d}\boldsymbol{x}$$
- For $\alpha = 2$:
$$D_2[p\|q] = \frac{1}{2}\int \frac{(p(\boldsymbol{x}) - q(\boldsymbol{x}))^2}{q(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x}$$
- Consider the KL-divergence to be:
$$\lim_{\alpha\to 0} D_\alpha[p\|q] = \mathrm{KL}[q\|p] \qquad \lim_{\alpha\to 1} D_\alpha[p\|q] = \mathrm{KL}[p\|q]$$

Remark 52. Local EP minimization gives fixed point update that blends message to power α with previous state approximation:
$$\tilde{f}_{\text{new}} = \arg\min_f \; \mathrm{KL}\left[f_i(\mathcal{Z}_i)^\alpha \tilde{f}_i(\mathcal{Z})^{1-\alpha} q_{\neg i}(\mathcal{Z}) \middle\| f(\mathcal{Z}_i) q_{\neg i}(\mathcal{Z})\right]$$
Given a small change (like $\alpha < 1$) leads to more stable update and reliable convergence.

6 Belief Propagation: Interpretation

6.1 Introduction

Definition 6.1. (Loopy Propagation) The joint distribution for *any* graph is given by:
$$P(\mathcal{X}) = \frac{1}{Z}\prod_{\text{nodes } i} f_i(\boldsymbol{x}_i) \prod_{\text{edges } (ij)} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$$
Message computed recusively with few guarantee of convergence as we have the following message:
$$M_{j\to i} = \sum_{\boldsymbol{x}_j} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_j(\boldsymbol{x}_j) \prod_{l\in \mathrm{ne}(j)\setminus\{i\}} M_{l\to j}(\boldsymbol{x}_j)$$
The marginal distribution are approximation in general:
$$P(\boldsymbol{x}_i) \approx b_i(\boldsymbol{x}_i) \propto f_i(\boldsymbol{x}_i) \prod_{k\in\mathrm{ne}(i)} M_{k\to i}(\boldsymbol{x}_i)$$
$$P(\boldsymbol{x}_i, \boldsymbol{x}_j) \approx b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \propto f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_i(\boldsymbol{x}_i) f_j(\boldsymbol{x}_j) \prod_{k\in\mathrm{ne}(i)\setminus\{j\}} M_{k\to i}(\boldsymbol{x}_i) \prod_{l\in\mathrm{ne}(j)\setminus\{i\}} M_{l\to j}(\boldsymbol{x}_j)$$

Remark 53. **(Dealing with Loops)** There are various way to deal with loop as we have:

- The belief propagation posterior marginal are approximate on all non-tree because over-counted, but converged approximate are frequently found to be good.
- Converge can be seen in: Tree, Graph with single step, Distribution with weak iteraction, Graph with long (and weak) loops, and Gaussian network (variance may also converged).
- Damping, as it is a common approach to encorate of EP:
$$M_{i\to j}^{\text{new}}(\boldsymbol{x}_j) = (1-\alpha) M_{i\to j}^{\text{old}} + \alpha \sum_{\boldsymbol{x}_i} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_i(\boldsymbol{x}_i) \prod_{k\in\mathrm{ne}(i)\setminus\{j\}} M_{k\to i}(\boldsymbol{x}_i)$$
- Variable can be groupped into cliques to improve accuracy: region graph approximate, cluster variable method, and junction graph.

6.2 Message Based EP

Proposition 6.1. ***(Loopy BP as Message-Based EP)*** *One can consider the connection between message-based EP and loopy BP, as they are equivalent.*

Proof. Consider the appoximate pairwise factor $\tilde{f}_{ij}$ as product of messages:

$$f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \approx \tilde{f}_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) = M_{i\to j}(\boldsymbol{x}_j) M_{j\to i}(\boldsymbol{x}_i)$$

Consider the approximation of the factorized distribution:

$$\begin{aligned} P(\mathcal{X}) &\approx \frac{1}{Z} \prod_{\text{nodes}(i)} f_i(\boldsymbol{x}_i) \prod_{\text{edges}(ij)} \tilde{f}_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ &= \frac{1}{Z} \prod_{\text{nodes}(i)} \left(f_i(\boldsymbol{x}_i) \prod_{j\in\mathcal{N}(i)} M_{j\to i}(\boldsymbol{x}_i) \right) = \prod_{\text{nodes}(i)} b_i(\boldsymbol{x}_i) \end{aligned}$$

with multiple factors for $\boldsymbol{x}_i$, which we consider the update on EP to be:

- *Deletion*: Consider the following $P(\mathcal{X})$ as we have:

$$\begin{aligned} P_{\neg ij}(X_i, X_j) &= \sum_{c\neq i,j} \frac{P(\mathcal{X})}{\tilde{f}(X_i, X_j)} = \sum_{c\neq i,j} \frac{P(\mathcal{X})}{M_{i\to j}(\boldsymbol{x}_j) M_{j\to i}(\boldsymbol{x}_i)} \\ &= \frac{1}{\tilde{f}(X_i, X_j)} \sum_{c\neq i,j} f_i(\boldsymbol{x}_i) f_j(\boldsymbol{x}_j) \prod_{k\in\mathcal{N}(i)} M_{k\to i}(\boldsymbol{x}_i) \prod_{l\in\mathcal{N}(j)} M_{l\to j}(\boldsymbol{x}_j) \left(\prod_{s\neq i,j} f_s(\boldsymbol{x}_s) \prod_{t\in\mathcal{N}(s)} M_{t\to s}(\boldsymbol{x}_s) \right) \\ &= f_i(\boldsymbol{x}_i) f_j(\boldsymbol{x}_j) \prod_{k\in\mathcal{N}(i)\setminus j} M_{k\to i}(\boldsymbol{x}_i) \prod_{l\in\mathcal{N}(j)\setminus i} M_{l\to j}(\boldsymbol{x}_j) \sum_{c\neq i,j} \left(\prod_{s\neq i,j} f_s(\boldsymbol{x}_s) \prod_{t\in\mathcal{N}(s)} M_{t\to s}(\boldsymbol{x}_s) \right) \\ &= f_i(\boldsymbol{x}_i) f_j(\boldsymbol{x}_j) \prod_{k\in\mathcal{N}(i)\setminus j} M_{k\to i}(\boldsymbol{x}_i) \prod_{l\in\mathcal{N}(j)\setminus i} M_{l\to j}(\boldsymbol{x}_j) \end{aligned}$$

- *Projection*: We consider minimizing the KL-divergence as we have:

$$\left\{ M^{\text{new}}_{i\to j}, M^{\text{new}}_{j\to i} \right\} = \underset{M_{i\to j}, M_{j\to i}}{\arg\min} \ \text{KL} \left[f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) q_{\neg ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \middle\| M_{j\to i}(\boldsymbol{x}_i) M_{i\to j}(\boldsymbol{x}_j) q_{\neg ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \right]$$

  To solve this KL-divergence, this is obvious, as $q_{\neg ij}(\cdot)$ can be factorized and so the minimizer is the marginal between $f_{ij}(\cdot) q_{\neg ij}(\cdot)$, which means that:

$$\begin{aligned} M^{\text{new}}_{i\to j}(\boldsymbol{x}_j) q_{\neg ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \sum_{\boldsymbol{x}_j} \left(f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_j(\boldsymbol{x}_j) \prod_{l\in\mathcal{N}(j)\setminus i} M_{l\to j}(\boldsymbol{x}_j) \right) \underbrace{f_i(\boldsymbol{x}_i) \prod_{k\in\mathcal{N}(i)\setminus j} M_{k\to i}(\boldsymbol{x}_i)}_{q_{\neg ij}(\boldsymbol{x}_i)} \\ \implies M^{\text{new}}_{i\to j}(\boldsymbol{x}_j) &= \sum_{\boldsymbol{x}_j} \left(f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_j(\boldsymbol{x}_j) \prod_{l\in\mathcal{N}(j)\setminus i} M_{l\to j}(\boldsymbol{x}_j) \right) \end{aligned}$$

  This is the Loopy BP update, and so both of the are equivalent.

□

Remark 54. **(Comments on the Loopy BP and EP)** There are some observation that we can make in the equivalent between loopy BP and EP algorithm:

- Unlike EP, this message based EP doesn't need 2 separate approximate as we have in the normal EP.
- This message based EP is loopy graph can be seen as a more constraint on approximate site and not just exponential family factor but the product of exponential family message.
- On a tree, message forward EP finds the same marginal as standard EP as the messages are calculated the same way. Similarly, the pairwise marginal can be found after converge by compute $\tilde{P}(z_{i-1}, z_i)$
- Factorization still remain valid even when original site lies in the appoximation exponential family already, so the loopy BP can be seen as form of EP.
- This doesn't help us with understanding the convergence property of EP.

6.3 Reparameterized on Tree

Remark 55. **(Tree-Based Representation)** We consider the joint factorization, which can be represented:

$$\begin{aligned} P(\mathcal{X}) &= \frac{1}{Z} \prod_{\text{nodes}(i)} f_i(\boldsymbol{x}_i) \prod_{\text{edges}(ij)} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \qquad \text{(Undirected Tree)} \\ &= P(\boldsymbol{x}_i) \prod_{i \neq r} P(\boldsymbol{x}_i | \boldsymbol{x}_{\text{pa}(i)}) \qquad \text{(Directed Rooted Tree)} \\ &= \prod_{\text{nodes}(i)} P(\boldsymbol{x}_i) \prod_{\text{edges}(ij)} \frac{P(\boldsymbol{x}_i, \boldsymbol{x}_j)}{P(\boldsymbol{x}_i) P(\boldsymbol{x}_j)} \qquad \text{(Pairwise Marginal)} \end{aligned}$$

The last on requires that $\sum_{\boldsymbol{x}_j} P(\boldsymbol{x}_i, \boldsymbol{x}_j) = P(\boldsymbol{x}_i)$.

- The unidrected tree isn't unique as if we multiply the factor $f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ by $g(\boldsymbol{x}_i)$ and dividing $f_i(\boldsymbol{x}_i)$ by the same $g(\boldsymbol{x}_i)$ doesn't change the distribution.
- BP can be seen as iteractive replacement of $f_i(\boldsymbol{x}_i)$ by local marginal of $p_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ along with corresponding representation of $f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ (recall the Hugin propagation)
- Converged BP on a tree finds $P(\boldsymbol{x}_i)$ and $P(\boldsymbol{x}_i, \boldsymbol{x}_j)$ allowing up to transform the undirected tree to pairwise marginal.

Remark 56. **(Reparameterization in Tree)** To consider the tree based reparameterization, we want to transform the representation from undirected tree to pairwise marginal as:

$$\prod_{\text{nodes}(i)} f_i(\boldsymbol{x}_i) \prod_{\text{edges}(ij)} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \implies \prod_{\text{nodes}(i)} P(\boldsymbol{x}_i) \prod_{\text{edges}(ij)} \frac{P(\boldsymbol{x}_i, \boldsymbol{x}_j)}{P(\boldsymbol{x}_i) P(\boldsymbol{x}_j)}$$

We will define the $f_{ij}^0 = f_{ij}$, while the singleton factor to be $f_1^0 = p_1^0 = 1$, we consider the following update: The update is based on the fact that we will act on the factors *as if* it is actually representing the probabilities: We will consider such a procedure on a node that has 2 incoming messages.

- Starting with joint, where if we multiply it by adjacent factors we get $P(\boldsymbol{x}_i, \boldsymbol{x}_j)$ i.e

$$p^{(n)}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{1}{Z_{ij}^{(n)}} f_i^{(n-1)}(\boldsymbol{x}_i) f_{ij}^{(n-1)}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_j^{(n-1)}(\boldsymbol{x}_j)$$

- Finding the marginal, as we have:

$$f_i^{(n)}(\boldsymbol{x}_i) = p^{(n)}(\boldsymbol{x}_i) = \sum_{\boldsymbol{x}_j} p^{(n)}(\boldsymbol{x}_i, \boldsymbol{x}_j) = f_i^{(n-1)}(\boldsymbol{x}_i) \underbrace{\sum_{\boldsymbol{x}_j} f_{ij}^{(n-1)}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_j^{(n-1)}(\boldsymbol{x}_j)}_{M_{j \to i}}$$

- To keep the normalization correctly, we divide the message so that the update on one passing giving us normalized term:

$$f_{ij}^{(n)} = \frac{f_{ij}^{(n-1)}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{M_{j\to i}(\boldsymbol{x}_j)}$$

- We now consider the next step with the next incoming message from node k to node i:

$$\begin{aligned} p^{(n)}(\boldsymbol{x}_i, \boldsymbol{x}_k) &= \frac{1}{Z_{ik}^{(n)}} f_i^{(n)}(\boldsymbol{x}_i) f_{ik}^{(n-1)}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_k^{(n-1)}(\boldsymbol{x}_j) \\ &= \frac{1}{Z_{ik}^{(n)}} f_i^{(n-1)}(\boldsymbol{x}_i) M_{j\to i}(\boldsymbol{x}_i) f_{ik}^{(n-1)}(\boldsymbol{x}_i, \boldsymbol{x}_k) f_k^{(n-1)}(\boldsymbol{x}_k) \end{aligned}$$

- Finding the singleton factor by marginalization

$$f_i^{(n)}(\boldsymbol{x}_i) = f_i^{(n-1)}(\boldsymbol{x}_i) M_{j\to i}(\boldsymbol{x}_i) \underbrace{\sum_{\boldsymbol{x}_k} f_{ik}^{(n-1)}(\boldsymbol{x}_i, \boldsymbol{x}_k) f_k^{(n-1)}(\boldsymbol{x}_k)}_{M_{k\to i}}$$

- And so, the normalization correction on the joint factor is:

$$f_{ik}^{(n)} = \frac{f_{ik}^{(n-1)}(\boldsymbol{x}_i, \boldsymbol{x}_k)}{M_{k\to i}(\boldsymbol{x}_i)}$$

We perform this update throughout the tree, which we do it in forward (e.g $i \to j$) and backward (e.g $j \to i$) manner, which gives us:

$$\begin{aligned} f_i^{(\infty)}(\boldsymbol{x}_i) &= \prod_{j\,\mathrm{ne}(i)} M_{j\to i}(\boldsymbol{x}_i) = P(\boldsymbol{x}_i) \\ f_{ij}^{(\infty)}(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \frac{f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{M_{j\to i}(\boldsymbol{x}_i) M_{i\to j}(\boldsymbol{x}_j)} \\ &= \frac{\prod_{k\in \mathrm{ne}(i)\setminus j} M_{k\to i} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \prod_{l\in \mathrm{ne}(j)\setminus i} M_{l\to i}(\boldsymbol{x}_j)}{\prod_{k\in \mathrm{ne}(i)\setminus j} M_{k\to i} M_{j\to i}(\boldsymbol{x}_i) M_{i\to j}(\boldsymbol{x}_j) \prod_{l\in \mathrm{ne}(j)\setminus i} M_{l\to i}(\boldsymbol{x}_j)} \\ &= \frac{\prod_{k\in \mathrm{ne}(i)\setminus j} M_{k\to i} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \prod_{l\in \mathrm{ne}(j)\setminus i} M_{l\to i}(\boldsymbol{x}_j)}{\prod_{k\in \mathrm{ne}(i)} M_{k\to i} \prod_{l\in \mathrm{ne}(j)} M_{l\to i}(\boldsymbol{x}_j)} \\ &= \frac{P(\boldsymbol{x}_i, \boldsymbol{x}_j)}{P(\boldsymbol{x}_i) P(\boldsymbol{x}_j)} \end{aligned}$$

the equation follows from the result from belief propagation.This kind of reparameterization allows us to avoid double counting, which is essentially a book-keeping method, espescially the normalizing part (there will be a case where the factor cancel with unnecessary message from singleton factor, as intended).

Remark 57. **(Comments on the BP on non-tree)** If this converges in a non-tree setting, then we have locally consistent belief i.e:

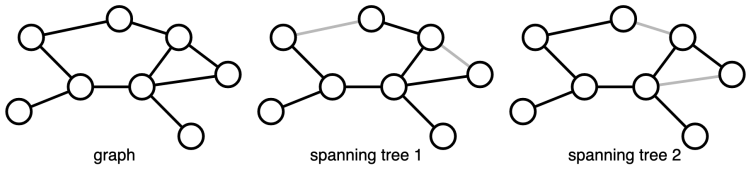
$$p(\mathcal{X}) \propto \prod_i b(\boldsymbol{x}_i) \prod_{ij} \frac{b(\boldsymbol{x}_i, \boldsymbol{x}_j)}{b(\boldsymbol{x}_i) b(\boldsymbol{x}_j)} \quad \text{such that} \quad \sum_{\boldsymbol{x}_j} b(\boldsymbol{x}_i, \boldsymbol{x}_j) = b(\boldsymbol{x}_i)$$

But it doesn't need to be globally consistent:

$$\sum_{\mathcal{X}_{\neg i}} \left(\prod_i b(\boldsymbol{x}_i) \prod_{ij} \frac{b(\boldsymbol{x}_i, \boldsymbol{x}_j)}{b(\boldsymbol{x}_i) b(\boldsymbol{x}_j)} \right) \neq b(\boldsymbol{x}_i)$$

This kind of marginal is called *pseudo-marginal*.

Remark 58. **(Message Schedule Scheme)** We consider update the belief on each *subtree* of the graph and passing message on each subtree, looping through all the subtree until converge:



And, we now that the following updates steps:

$$
\begin{aligned}
P(\mathcal{X}) &= \frac{1}{Z} \prod_{\text{nodes}(i)} f_i^{(0)}(\boldsymbol{x}_i) \prod_{\text{edges}(ij)} f_{ij}^{(0)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \\
&= \frac{1}{Z} \prod_{\text{nodes}(i) \in T_1} f_i^{(0)}(\boldsymbol{x}_i) \prod_{\text{edges}(ij) \in T_1} f_{ij}^{(0)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \left(\prod_{\text{edges}(ij) \notin T_1} f_{ij}^{(0)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \right) \\
&\quad \text{(Update)} \\
&= \frac{1}{Z} \prod_{\text{nodes}(i) \in T_1} f_i^{(1)}(\boldsymbol{x}_i) \prod_{\text{edges}(ij) \in T_1} f_{ij}^{(1)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \left(\prod_{\text{edges}(ij) \notin T_1} f_{ij}^{(1)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \right) \\
&\quad \text{(Next Tree)} \\
&= \frac{1}{Z} \prod_{\text{nodes}(i) \in T_2} f_i^{(1)}(\boldsymbol{x}_i) \prod_{\text{edges}(ij) \in T_2} f_{ij}^{(1)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \left(\prod_{\text{edges}(ij) \notin T_2} f_{ij}^{(1)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \right) \\
&\dots
\end{aligned}
$$

where we have, when we got a new tree,

$$
f_i^{(1)}(\boldsymbol{x}_i) = P^{T_1}(\boldsymbol{x}_i) \qquad f_{ij}^{(1)}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{P^{T_1}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{P^{T_1}(\boldsymbol{x}_i) P^{T_2}(\boldsymbol{x}_j)}
$$

If the process converges, suppose it converge to:

$$
P(\mathcal{X}) = \frac{1}{Z} \prod_{\text{nodes}(i)} f_i^{(\infty)}(\boldsymbol{x}_i) \prod_{\text{edges}(ij)} f_{ij}^{(\infty)}(\boldsymbol{x}_i, \boldsymbol{x}_j)
$$

where for any tree T in the graph, we have:

$$
f_i^{(\infty)} = P^T(\boldsymbol{x}_i) \qquad f_{ij}^{(\infty)} = \frac{P^T(\boldsymbol{x}_i, \boldsymbol{x}_j)}{P^T(\boldsymbol{x}_i) P^T(\boldsymbol{x}_j)}
$$

This means that the local marginal of all subtree are consistent with each other, and the pseudo-marginal is valid belief of any of the subtree, as this is stronger constriant.

6.4 Bathe Free Energy

Remark 59. **(Introduction to Bathe Free Energy)** In reparameterization view, BP solves for marginal belief $b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and $b_i(\boldsymbol{x}_i) = \sum_{\boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ such that:

$$
P(\mathcal{X}) \propto \prod_i f_i(\boldsymbol{x}_i) \prod_{ij} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \propto \prod_i b_i(\boldsymbol{x}_i) \prod_{ij} \frac{b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{b_i(\boldsymbol{x}_i) b_j(\boldsymbol{x}_j)}
$$

Loopy BP is a set of fixed point equation for finding stationary of an objective function called Bathe free energy, which is defined in terms of locally consistent belief (pseudo-marginal) $b_i \geq 0$ and $b_{ij} \geq 0$ such that:

$$\sum_{\boldsymbol{x}_i} b_i(\boldsymbol{x}_i) = 1 \qquad \sum_{\boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) = b_i(\boldsymbol{x}_i)$$

Definition 6.2. (Bathe Free Energy) We define it in the form of:

$$\mathcal{F}_{\text{bathe}}(b) = \mathcal{E}_{\text{bathe}}(b) + \mathcal{H}_{\text{bathe}}(b)$$

Both terms are approximated so that it corresponds to variational likelihood terms:

- Bathe average energy is the expected log-joint evaluate as though the pseudomarginal were correct:

$$\mathcal{E}_{\text{bathe}}(b) = \sum_i \sum_{\boldsymbol{x}_i} b_i(\boldsymbol{x}_i) \log f_i(\boldsymbol{x}_i) + \sum_{ij} \sum_{\boldsymbol{x}_i \boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \log f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

- Bathe entropy is the sum of pseudomarginal entropies corrected for pairwise (pseudo-)interaction, but neglecting higher-order dependence:

$$\begin{aligned}\mathcal{H}_{\text{bathe}}(b) &= \sum_i H[b_i] - \sum_{ij} \text{KL}[b_{ij} | b_i b_j] \\ &= -\sum_i \sum_{\boldsymbol{x}_i} b_i(\boldsymbol{x}_i) \log b_i(\boldsymbol{x}_i) - \sum_{ij} \sum_{\boldsymbol{x}_i \boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \log \frac{b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{b_i(\boldsymbol{x}_i) b_j(\boldsymbol{x}_j)}\end{aligned}$$

On tree, both belief and the bathe entropy expression are correct i.e $\mathcal{F}_{\text{bathe}} = \mathcal{F}$. The update rule can be recoved from finding the fixed point.

Proposition 6.2. ***(Fixed Point for Bathe Free Energy)*** *The fixed point for Bathe free energy is:*

$$\begin{aligned}b_i(\boldsymbol{x}_i) &\propto f_i(\boldsymbol{x}_i) \prod_{j \in \text{ne}(i)} \exp(-\xi_{ij}(\boldsymbol{x}_i)) \\ b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) &\propto f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) b_i(\boldsymbol{x}_i) b_j(\boldsymbol{x}_j) \exp(\xi_{ij}(\boldsymbol{x}_i) + \xi_{ij}(\boldsymbol{x}_j)) \\ \exp(-\xi_{ij}(\boldsymbol{x}_i)) &\propto \sum_{\boldsymbol{x}_j} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_j(\boldsymbol{x}_j) \prod_{l \in \text{ne}(j) \setminus i} \exp(-\xi_{ij}(\boldsymbol{x}_j))\end{aligned}$$

Proof. We find the Lagragian with local consistency and normalization, which is given as:

$$\begin{aligned}\mathcal{L} = &\sum_i \sum_{\boldsymbol{x}_i} b_i(\boldsymbol{x}_i) \log f_i(\boldsymbol{x}_i) + \sum_{ij} \sum_{\boldsymbol{x}_i \boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \log f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ &- \sum_i \sum_{\boldsymbol{x}_i} b_i(\boldsymbol{x}_i) \log b_i(\boldsymbol{x}_i) - \sum_{ij} \sum_{\boldsymbol{x}_i \boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \log \frac{b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{b_i(\boldsymbol{x}_i) b_j(\boldsymbol{x}_j)} \\ &+ \sum_i \xi_i \left(\sum_{\boldsymbol{x}_i} b_i(\boldsymbol{x}_i) - 1 \right) \\ &+ \sum_{ij} \left[\sum_{\boldsymbol{x}_i} \xi_{ij}(\boldsymbol{x}_i) \left(\sum_{\boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) - b_i(\boldsymbol{x}_i) \right) + \sum_{\boldsymbol{x}_j} \xi_{ij}(\boldsymbol{x}_j) \left(\sum_{\boldsymbol{x}_i} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) - b_j(\boldsymbol{x}_j) \right) \right]\end{aligned}$$

Setting the derivate to zero, which gives us the solution:

$$\begin{aligned}\frac{\partial}{\partial b_i(\boldsymbol{x}_i)} &= \log f_i(\boldsymbol{x}_i) - \log b_i(\boldsymbol{x}_i) + \sum_{j \in \text{ne}(j)} \sum_{\boldsymbol{x}_j} \frac{b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{b_i(\boldsymbol{x}_i)} + \xi_i - \sum_{j \in \text{ne}(i)} \xi_{ij}(\boldsymbol{x}_i) + \text{const.} = 0 \\ &\implies b_i(\boldsymbol{x}_i) \propto f_i(\boldsymbol{x}_i) \prod_{j \in \text{ne}(i)} \exp(-\xi_{ij}(\boldsymbol{x}_i))\end{aligned}$$

$$\frac{\partial f}{\partial b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)} = \log f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) - \log b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) + \log b_i(\boldsymbol{x}_i) b_j(\boldsymbol{x}_j) + \xi_{ij}(\boldsymbol{x}_i) + \xi_{ji}(\boldsymbol{x}_j) + \text{const.} = 0$$
$$\implies b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \propto f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) b_i(\boldsymbol{x}_i) b_j(\boldsymbol{x}_j) \exp(\xi_{ij}(\boldsymbol{x}_i) + \xi_{ji}(\boldsymbol{x}_j))$$

To solve for $\xi_{ij}(\boldsymbol{x}_i)$ by enforcing the constant $\sum_{\boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) = b_i(\boldsymbol{x}_i)$ where we have:

$$\sum_{\boldsymbol{x}_j} b_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \propto \sum_{\boldsymbol{x}_j} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) b_i(\boldsymbol{x}_i) b_j(\boldsymbol{x}_j) \exp(\xi_{ij}(\boldsymbol{x}_i) + \xi_{ji}(\boldsymbol{x}_j))$$
$$\implies b_i(\boldsymbol{x}_i) \propto b_i(\boldsymbol{x}_i) \exp(\xi_{ij}(\boldsymbol{x}_i)) \sum_{\boldsymbol{x}_j} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) b_j(\boldsymbol{x}_j) \exp(\xi_{ji}(\boldsymbol{x}_j))$$
$$\implies \exp(-\xi_{ij}(\boldsymbol{x}_i)) \propto \sum_{\boldsymbol{x}_j} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) b_j(\boldsymbol{x}_j) \exp(\xi_{ji}(\boldsymbol{x}_j))$$
$$= \sum_{\boldsymbol{x}_j} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) f_j(\boldsymbol{x}_j) \prod_{l \in \text{ne}(j) \setminus i} \exp(-\xi_{ji}(\boldsymbol{x}_j))$$

□

Remark 60. **(Interpretation of Results)** Comparing with BP, we have the message to be of the form of $M_{j \to i}(\boldsymbol{x}_i) = \exp(-\xi_{ij}(\boldsymbol{x}_i))$. The fixed point for bathe free energy recovers the message passing rule:

- Stable Fixed point of loopy BP are stationary point of Bathe and local minimum of Bathe free energy.
- For binary attractive netwrok: the Bathe free energy at fixed point of loopy BP provides an upperbound on the log partition function $\log P(\boldsymbol{Z})$.
- It is useful for learning undirected graphical model as it leads to lower bound on the log-likelihood.
- Belief b_i and b_{ij} in loopy BP are only locally consistent pseudomarginal, not necessary consistent with marginal or the implied joint distribution.
- Bathe free enerfy accounts for interaction between difference states, while variational free energy that assume independence.
- The log series Plefka expansion of the log-partition Z: the variational energy form the first order while Bathe free energy contains higher term.
- Loopy BP tends to significantly more accurate whenever it converges.

Remark 61. **(Extensions and Variations)**

- Generalized BP is a group variable together to threat their interaction exacely.
- The algorithm can be derived so that the Bathe free energy at every step and thus guarantee the convergence. Similarly, convex alternative and we will converge to unique global maximum.
- The treatment of loopy Viterbi or max-product algorithm is difference.

7 Exponential families: Convexity, Duality and Free Energies

Definition 7.1. (Log-Partition Function) Consider the exponential family distribution with sufficient statistics $\boldsymbol{s}(\boldsymbol{x})$ and natural parameter $\boldsymbol{\theta}$. We have the following probability to be $P(\boldsymbol{x}|\boldsymbol{\theta}) = \exp(\boldsymbol{\theta}^T \boldsymbol{s}(\boldsymbol{x}) - \boldsymbol{\Phi}(\boldsymbol{\theta}))$, where $\boldsymbol{\Phi}(\boldsymbol{\theta})$ is the log-partition, where:

$$\boldsymbol{\Phi}(\boldsymbol{\theta}) = \log \int \exp(\boldsymbol{\theta}^T \boldsymbol{s}(\boldsymbol{x})) \, \mathrm{d}\boldsymbol{x}$$

Proposition 7.1. ***(Derivative of Log Partitions)*** *We can show that:*

$$\frac{\partial}{\partial \boldsymbol{\theta}}\boldsymbol{\Phi}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})}[\boldsymbol{s}(\boldsymbol{x})] = \boldsymbol{\mu}(\boldsymbol{\theta})$$
$$\frac{\partial^2}{\partial \boldsymbol{\theta}^2}\boldsymbol{\Phi}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})}[\boldsymbol{s}(\boldsymbol{x})^2] - \mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})}[\boldsymbol{s}(\boldsymbol{x})^2] = \mathbb{V}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})}[\boldsymbol{s}(\boldsymbol{x})]$$

The second derivative is positive semi-definite and so $\boldsymbol{\Phi}(\boldsymbol{\theta})$ is convex in $\boldsymbol{\theta}$.

Proof. The first result is the old results. Now, consider the second derivation as we have:

$$\begin{aligned}
\frac{\partial^2}{\partial^2 \boldsymbol{\theta}}\boldsymbol{\Phi}(\boldsymbol{\theta}) &= \frac{\partial}{\partial \boldsymbol{\theta}}\left[\exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\int \boldsymbol{s}(\boldsymbol{x})\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}\right] \\
&= \left[\frac{\partial}{\partial \boldsymbol{\theta}}\exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\right]\int \boldsymbol{s}(\boldsymbol{x})\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x} \\
&\qquad + \exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\left[\frac{\partial}{\partial \boldsymbol{\theta}}\int \boldsymbol{s}(\boldsymbol{x})\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}\right] \\
&= -\exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\left[\frac{\partial}{\partial \boldsymbol{\theta}}\boldsymbol{\Phi}(\boldsymbol{\theta})\right]\int s(\boldsymbol{x})\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x} \\
&\qquad + \exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\left[\frac{\partial}{\partial \boldsymbol{\theta}}\int \boldsymbol{s}(\boldsymbol{x})\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}\right] \\
&= -\exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\left[\frac{\partial}{\partial \boldsymbol{\theta}}\boldsymbol{\Phi}(\boldsymbol{\theta})\right]\int \boldsymbol{s}(\boldsymbol{x})\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x} \\
&\qquad + \exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\left[\int \boldsymbol{s}(\boldsymbol{x})^2\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}\right] \\
&= -\left[\exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\int \boldsymbol{s}(\boldsymbol{x})\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}\right]^2 + \exp(-\boldsymbol{\Phi}(\boldsymbol{\theta}))\left[\int \boldsymbol{s}(\boldsymbol{x})^2\exp(\boldsymbol{\theta}^T\boldsymbol{s}(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}\right] \\
&= -\mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})}[\boldsymbol{s}(\boldsymbol{x})^2] + \mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})}[\boldsymbol{s}(\boldsymbol{x})^2]
\end{aligned}$$

□

Definition 7.2. (Convex Duality/Conjugate) Given a function $f^*(\boldsymbol{x})$, we denote the duality/conjugate of the function is:

$$f(\boldsymbol{y}) = \sup_{\boldsymbol{x}\in\mathcal{X}}\Big(\boldsymbol{y}^T\boldsymbol{x} - f(\boldsymbol{x})\Big)$$

Definition 7.3. (Negative Entropy of Mean parameter) The negative entropy of the distribution as a function of mean parameter:

$$\Psi(\boldsymbol{\mu}) = \mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})}[\log P(\boldsymbol{x}|\boldsymbol{\theta})] = \boldsymbol{\theta}^T\mathbb{E}[\boldsymbol{s}(\boldsymbol{x})] - \boldsymbol{\Phi}(\boldsymbol{\theta}) = \boldsymbol{\theta}^T\boldsymbol{\mu} - \boldsymbol{\Phi}(\boldsymbol{\theta})$$

Note that the variable $\boldsymbol{\mu}$ is arbitary:

Lemma 7.1. *We can observe that the derivative of negative entropy recovers the natural parameter*

$$\frac{d}{d\boldsymbol{\mu}}\boldsymbol{\Psi}(\boldsymbol{\mu}) = \boldsymbol{\theta}$$

This related to dual function.

Proof. We have the following distribution:

$$\begin{aligned}
\frac{d}{d\boldsymbol{\mu}}\boldsymbol{\Psi}(\boldsymbol{\mu}) &= \frac{\partial}{\partial \boldsymbol{\mu}}(\boldsymbol{\theta}^T\boldsymbol{\mu} - \boldsymbol{\Phi}(\boldsymbol{\theta})) + \frac{d\boldsymbol{\theta}}{d\boldsymbol{\mu}}\frac{\partial}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}^T\boldsymbol{\mu} - \boldsymbol{\Psi}(\boldsymbol{\theta})) \\
&= \boldsymbol{\theta} + \frac{d\boldsymbol{\theta}}{d\boldsymbol{\theta}}(\boldsymbol{\mu} - \boldsymbol{\mu}) = \boldsymbol{\theta}
\end{aligned}$$

□

Proposition 7.2. ***(Duality Between Partition and Entropy)*** *One can show that partition and entroy are duality of each other:*

$$\boldsymbol{\Psi}(\boldsymbol{\mu}) = \sup_{\boldsymbol{\theta}'} \Big[(\boldsymbol{\theta}')^T \boldsymbol{\mu} - \boldsymbol{\Phi}(\boldsymbol{\theta}') \Big] \qquad \boldsymbol{\Phi}(\boldsymbol{\theta}) = \sup_{\boldsymbol{\mu}'} \Big[\boldsymbol{\theta}^T \boldsymbol{\mu}' - \boldsymbol{\Psi}(\boldsymbol{\mu}') \Big]$$

Proof. Consider the KL-divergence between distribution with natural parameter $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ as we have:

$$\begin{aligned}
\mathrm{KL}[p\|q] &= \mathrm{KL}\Big[P(\boldsymbol{x}|\boldsymbol{\theta}) \Big\| P(\boldsymbol{x}|\boldsymbol{\theta}') \Big] \\
&= \mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})} \Big[-\log P(\boldsymbol{x}|\boldsymbol{\theta}') + \log P(\boldsymbol{x}|\boldsymbol{\theta}) \Big] \\
&= \mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})} \Big[-\log P(\boldsymbol{x}|\boldsymbol{\theta}') \Big] + \mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x}|\boldsymbol{\theta})} \Big[\log P(\boldsymbol{x}|\boldsymbol{\theta}) \Big] \\
&= -(\boldsymbol{\theta}')^T \boldsymbol{\mu} + \boldsymbol{\Phi}(\boldsymbol{\theta}') + \boldsymbol{\Psi}(\boldsymbol{\mu}) \geq 0
\end{aligned}$$

We can see that $\boldsymbol{\Psi}(\boldsymbol{\mu}) \geq (\boldsymbol{\theta}')^T \boldsymbol{\mu} - \boldsymbol{\Phi}(\boldsymbol{\theta}')$. With the minimization of KL to be $\boldsymbol{\theta} = \boldsymbol{\theta}'$, and so we have the first dual formula, while the second formula comes from the dual property (dual of dual is the function itself). □

Remark 62. **(Another Proof)** One can consider the sup by finding derivative and set it to zero:

$$\frac{\partial}{\partial \boldsymbol{\theta}'} \Big[(\boldsymbol{\theta}')^T \boldsymbol{\mu} - \boldsymbol{\Phi}(\boldsymbol{\theta}') \Big] = \boldsymbol{\mu} - \boldsymbol{\mu}(\boldsymbol{\theta}) = 0$$

Setting to zero and we have that $\boldsymbol{\mu} = \boldsymbol{\mu}(\boldsymbol{\theta})$, plugging it back and we get the negative entropy. This is the same as the second formula as:

$$\frac{\partial}{\partial \boldsymbol{\mu}'} \Big[\boldsymbol{\theta}^T \boldsymbol{\mu}' - \boldsymbol{\Psi}(\boldsymbol{\mu}') \Big] = \boldsymbol{\theta} - \boldsymbol{\theta}(\boldsymbol{\mu}) = 0$$

Note that $\boldsymbol{\theta}(\boldsymbol{\mu})$ doesn't have a full definition but we can consider it from the definition of negative entropy. Thus, complete the proof.

Remark 63. **(Free Energy and Dual)** We have the following components:

- Joint exponential family distribution of observed $\boldsymbol{x}$ and $\boldsymbol{z}$:

$$P(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}) = \exp\Big[\boldsymbol{\theta}^T \boldsymbol{s}(\boldsymbol{x}, \boldsymbol{z}) - \Phi_{xz}(\boldsymbol{\theta}) \Big]$$

- The posterior $\boldsymbol{z}$ is in exponential family with clamped sufficient statistics:

$$P(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{\theta}) = \exp\Big[\boldsymbol{\theta}^T \boldsymbol{s}_z(\boldsymbol{z}; \boldsymbol{x}) - \Phi_z(\boldsymbol{\theta}) \Big]$$

  where we have $s_z(\boldsymbol{z}; \boldsymbol{x}) = \boldsymbol{s}_{xz}(\boldsymbol{x}^{\text{obs}}; \boldsymbol{z})$

We consider the likelihood (of observed variables) to be as:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\theta}) = P(\boldsymbol{x}|\boldsymbol{\theta}) &= \int \exp\Big[\boldsymbol{\theta}^T \boldsymbol{s}(\boldsymbol{x}, \boldsymbol{z}) - \Phi_{xz}(\boldsymbol{\theta}) \Big] \, \mathrm{d}\boldsymbol{z} \\
&= \int \exp\Big[\boldsymbol{\theta}^T \boldsymbol{s}(\boldsymbol{x}, \boldsymbol{z}) \Big] \exp\Big[-\Phi_{xz}(\boldsymbol{\theta}) \Big] \, \mathrm{d}\boldsymbol{z} \\
&= \exp\Big[-\Phi_{xz}(\boldsymbol{\theta}) \Big] \int \exp\Big[\boldsymbol{\theta}^T \boldsymbol{s}(\boldsymbol{z}; \boldsymbol{x}) \Big] \, \mathrm{d}\boldsymbol{z} \\
&= \exp\Big[-\Phi_{xz}(\boldsymbol{\theta}) \Big] \exp\Big[\Phi_z(\boldsymbol{\theta}) \Big]
\end{aligned}$$

We can see that the log-likelihood is given as:

$$\begin{aligned}\log \mathcal{L}(\boldsymbol{\theta}) = l(\boldsymbol{\theta}) &= \Phi_z(\boldsymbol{\theta}) - \Phi_{xz}(\boldsymbol{\theta}) \\ &= \sup_{\boldsymbol{\mu}_z} \left[\boldsymbol{\theta}^T \boldsymbol{\mu}_z - \Psi(\boldsymbol{\mu}_z)\right] - \Phi_{xz}(\boldsymbol{\theta}) \\ &= \sup_{\boldsymbol{\mu}_z} \Big[\underbrace{\boldsymbol{\theta}^T \boldsymbol{\mu}_z - \Phi_{xz}(\boldsymbol{\theta})}_{\langle \log P(\boldsymbol{x},\boldsymbol{z})\rangle_q} - \underbrace{\Psi(\boldsymbol{\mu}_z)}_{-H[q]}\Big] = \sup_{\boldsymbol{\mu}_z} \mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\mu}_z)\end{aligned}$$

Please note that, we can see $\boldsymbol{\mu}_z = \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z})}$ for some variational distribution q (that we want to optimize). And so, one can recovere the free energy formulation from the duality of entropy and expected sufficient statistics.

Remark 64. **(Optimization of Free Energy)** We can see that the optimization over the $\boldsymbol{\mu}$ directly, as:

$$\boldsymbol{\mu}_z^* = \arg\max_{\boldsymbol{\mu}_z} \left[\boldsymbol{\theta}^T \boldsymbol{\mu}_z - \Psi(\boldsymbol{\mu})\right]$$

This is concave maximization; however, we have 2 complications to the problem:

- The optimization must be found out feasible means. Interdependence of the sufficient statistics may prevent arbitary sets of mean sufficient statistics being achieved.
- The feasible means are convex combination of all single configuration of sufficient statistics:

$$\boldsymbol{\mu} = \int \nu(\boldsymbol{x}) s(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \qquad\qquad \int \mu(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} = 1$$

- Let's consider the Boltzman machine on 2 variables, as we have:
  - Consider the Boltzman machine on 2 variables as we have x_1 and x_2:

$$E = \frac{1}{Z} \exp\left(-\sum_{i<j} w_{ij} x_i x_j + \sum_i \theta_i x_i\right)$$

  - Sufficient statistics of the Boltzman machine is: $\boldsymbol{s}(\boldsymbol{x}_1, \boldsymbol{x}_2) = [\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_1\boldsymbol{x}_2]$
  - There are only 4 states as we have: $\mathcal{S} = \{[0,0,0], [0,1,0], [1,0,0], [1,1,1]\}$
  - It is clear that $\boldsymbol{\mu}$ is in the convex hull of $\mathcal{S}$.
- For discrete distribution, the space of possible mean is bounded exponentially many hyperplanes connected the discrete configuration set called marginal polytrope.
- Even with marginal polytrope to marginal polytrope, evaluating $\boldsymbol{\Psi}(\boldsymbol{\mu})$ is challenging.

Remark 65. **(Undirected Tree and Markov Random Field)** We can parameterize a discrete MFR:

$$\begin{aligned}P(\mathcal{X}) &= \frac{1}{Z} \prod_i f_i(\boldsymbol{x}_i) \prod_{ij} f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ &= \exp\left(\sum_i \sum_k \theta_i(k)\delta(\boldsymbol{x}_i = k) + \sum_{ij} \sum_{k,l} \theta_{ij}(k,l)\delta(\boldsymbol{x}_i = k)\delta(\boldsymbol{x}_j = l) - \Phi(\boldsymbol{\theta})\right)\end{aligned}$$

Disrete MRF are always exponential family, with natural and mean parameter:

$$\boldsymbol{\theta} = \begin{bmatrix}\theta_i(k) & \theta_{ij}(k,l)\end{bmatrix} \quad \boldsymbol{\mu} = \begin{bmatrix}P(\boldsymbol{x}_i = k) & P(\boldsymbol{x}_i = k, \boldsymbol{x}_j = l)\end{bmatrix}$$

for all i, j, k, l. In particular, the mean parameter are just singleton and pairwise probability tables.

Remark 66. **(Entropy of MRF)** If the MRF has tree structure T and the negate entropy, can be written in term of single state entropy and mutual information on edges as (note the reparameterization and the tree structure):

$$\begin{aligned}\Psi(\boldsymbol{\mu}_T) &= \mathbb{E}_{\boldsymbol{x}_i \sim P(\boldsymbol{x}|\boldsymbol{\theta}_T)}\left[\log \prod_i P(\boldsymbol{x}_i) \prod_{(ij)\in \text{edge}(T)} \frac{P(\boldsymbol{x}_i, \boldsymbol{x}_j)}{P(\boldsymbol{x}_i)P(\boldsymbol{x}_j)}\right] \\ &= -\sum_i H(\boldsymbol{x}_i) + \sum_{(ij)\in \text{edge}(T)} I(\boldsymbol{x}_i, \boldsymbol{x}_j)\end{aligned}$$

Remark 67. **(Bathe Energy of Free Energy)** Let's consider the Bathe free energy, as a relaxation of the free-energy optimization, which is denoted as:

$$\boldsymbol{\mu}_z^* = \arg\max_{\boldsymbol{\mu}_z} \left[\boldsymbol{\theta}^T \boldsymbol{\mu}_z - \Psi(\boldsymbol{\mu}_z)\right]$$

Consider the set $\mathcal{M}$ as the set of feasible means. Now, we have the following relaxiation:

- *Feasible Set* Relax the $\mathcal{M}$ to be $\mathcal{L}$ be the set of locally consistent means (all nested means marginalized correctly), instead of being globally estimate.
- *Approximate* $\Psi(\boldsymbol{\mu}_z)$ is the entropy of an arbitary graph. However, it is hard to evaluate it exactly, and so we consider a result from tree structure but in the final calculation, we consider every edges:

$$\Psi_{\text{Bathe}}(\boldsymbol{\mu}_z) = -\sum_i H(\boldsymbol{x}_i) + \sum_{(ij)\in \text{edge}(G)} I(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

Please note that $\mathcal{L}$ is still a convex set. However, Ψ_{Bathe} isn't convex (Please recall the dual formulation as we can find Φ instead).

Remark 68. **(Upperbound on Log Partition)** Consider upperbound on $\Phi(\boldsymbol{\theta})$, imagine a set of spanning tree T for the MRF, with corresponding: $\boldsymbol{\theta}_T$ and $\boldsymbol{\mu}_T$ (can be done by padding the entries corresponding to off-tree edges with zero, and so $\boldsymbol{\theta}_T$ have the same size of $\boldsymbol{\theta}$). Consider the following:

- Consider the distribution over the spanning tree β as, we need $\mathbb{E}_\beta[\boldsymbol{\theta}_T] = \boldsymbol{\theta}$. Using the convexity of Φ:

$$\Phi(\boldsymbol{\theta}) = \Phi(\mathbb{E}_\beta[\boldsymbol{\theta}_T]) \leq \mathbb{E}_\beta[\Phi(\boldsymbol{\theta}_T)]$$

- The tighter upperbound can be obtained by minimizing Φ as we have:

$$\Phi(\boldsymbol{\theta}) \leq \inf_{\boldsymbol{\theta}_T : \mathbb{E}_\beta[\boldsymbol{\theta}_T] = \boldsymbol{\theta}} \mathbb{E}_\beta[\Phi(\boldsymbol{\theta}_T)] = \Phi_\beta(\boldsymbol{\theta})$$

- We can maximize the free-energy as we want (together with the Lagrange multiplier) and so, we have, the following optimization problem:

$$\sup_{\boldsymbol{\lambda}} \inf_{\boldsymbol{\theta}_T} \left(\mathbb{E}_\beta[\Phi(\boldsymbol{\theta}_T)] - \boldsymbol{\lambda}^T(\mathbb{E}_\beta[\boldsymbol{\theta}_T] - \boldsymbol{\theta})\right)$$

Proposition 7.3. ***(Solving Optimization - Tighter Bound)*** *Given the following optimization problem:*

$$\sup_{\boldsymbol{\lambda}} \inf_{\boldsymbol{\theta}_T} \left(\mathbb{E}_\beta[\Phi(\boldsymbol{\theta}_T)] - \boldsymbol{\lambda}^T(\mathbb{E}_\beta[\boldsymbol{\theta}_T] - \boldsymbol{\theta})\right)$$

This implies that $\boldsymbol{\mu}_T = \Pi_T(\boldsymbol{\lambda})$ as it is Lagrange multiplier corresponding to non-zero element of $\boldsymbol{\theta}$ for each tree (but the $\boldsymbol{\mu}$ stays the same (for all tree), see tree-reparametrisation view of BP). The Lagrange multiplier and $\Phi_\beta(\boldsymbol{\theta})$ can be find by:

$$\Phi_\beta(\boldsymbol{\theta}) = \sup_{\boldsymbol{\lambda}} \boldsymbol{\lambda}^T \boldsymbol{\theta} + \sum_i H_\lambda(\boldsymbol{\theta}) - \sum_{ij} \beta_{ij} I_\lambda(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

where β_{ij} is the probability distribution over tree.

Proof. Consider the derivative of Lagrange multiplier as we have:

$$\begin{aligned}
&\frac{\partial}{\partial \boldsymbol{\theta}_T} \sum_T \beta(T)\Phi(\boldsymbol{\theta}_T) - \boldsymbol{\lambda}^T \frac{\partial}{\partial \boldsymbol{\theta}_T} \sum_{\boldsymbol{T}} \beta(\boldsymbol{T})\boldsymbol{\theta}_T = 0 \\
\iff &\beta(T)\boldsymbol{\mu}_T - \beta(T)\Pi_T(\boldsymbol{\lambda}) = 0 \\
\iff &\boldsymbol{\mu}_T = \Pi_T(\boldsymbol{\lambda})
\end{aligned}$$

Thus the first point is proven. Now, consider the $\Phi_\beta(\boldsymbol{\beta})$ as we have:

$$\begin{aligned}
\Phi_\beta(\boldsymbol{\theta}) &= \sup_{\boldsymbol{\lambda}} \inf_{\boldsymbol{\theta}_T} \Big(\mathbb{E}_\beta[\Phi(\boldsymbol{\theta}_T)] - \boldsymbol{\lambda}^T (\mathbb{E}_\beta[\boldsymbol{\theta}_T] - \boldsymbol{\theta}) \Big) \\
&= \sup_{\boldsymbol{\lambda}} \boldsymbol{\lambda}^T \boldsymbol{\theta} + \mathbb{E}_\beta \left[\inf_{\boldsymbol{\theta}_T} \Big(\Phi(\boldsymbol{\theta}_T) - \boldsymbol{\theta}_T^T \Pi_T(\boldsymbol{\lambda}) \Big) \right] \\
&= \sup_{\boldsymbol{\lambda}} \boldsymbol{\lambda}^T \boldsymbol{\theta} + \mathbb{E}_\beta \Big[-\Psi(\Pi_T(\boldsymbol{\lambda})) \Big] \\
&= \sup_{\boldsymbol{\lambda}} \boldsymbol{\lambda}^T \boldsymbol{\theta} + \mathbb{E}_\beta \left[\sum_i H_\lambda(\boldsymbol{x}_i) - \sum_{(ij)\in T} I_\lambda(\boldsymbol{x}_i, \boldsymbol{x}_j) \right] \\
&= \sup_{\boldsymbol{\lambda}} \boldsymbol{\lambda}^T \boldsymbol{\theta} + \sum_i H_\lambda(\boldsymbol{x}_i) - \sum_{(ij)\in T} \beta_{ij} I_\lambda(\boldsymbol{x}_i, \boldsymbol{x}_j)
\end{aligned}$$

Thus the proposition is proven. □

Remark 69. **(Interpretation of Result)** This is the convexifaction of the Bethe free energy. We can optimization with respected to $\boldsymbol{\lambda}$, which is approximate inference applied to tightest bound on $\Phi(\boldsymbol{\theta})$ for a fixed β. The bound can be tighten by the optimization of β.

Remark 70. **(Bathe Free Energy and EP - Setup)** A Bathe-like approach also casts EP as variational energy fixed point method. Consider the marginals of a posterior distribution by cliques potential:

$$P(\mathcal{Z}) \propto f_0(\mathcal{Z}) \prod_i f_i(\mathcal{Z})$$

There are some remarks on each of the components:

- All factors are exponential form
- f_0 is tractable exponential family (possible uniform)
- f_i are jointly intractable, but the product can't be marginalized, although individual terms maybe tractable.

We consider the tractable exponential family terms with zero natural parameter together with the tractable sufficient statistics $\tilde{s}(\mathcal{Z}_i)$ as (can be seen as an approximating sites):

$$P(\mathcal{Z}) = \exp\Big(\boldsymbol{\theta}_0^T \boldsymbol{s}_0(\mathcal{X})\Big) \prod_i \exp\Big(\boldsymbol{\theta}_i^T \boldsymbol{s}_i(\mathcal{Z}_i)\Big) \exp\big(\mathbf{0}^T \tilde{\boldsymbol{s}}(\mathcal{Z}_i)\big)$$

Now, we can consider any given natural parameter $\tilde{\boldsymbol{\theta}}$ but it will be initialized with $\mathbf{0}$ (it can change):

$$P(\mathcal{Z}) = \exp\left(\boldsymbol{\theta}_0^T \boldsymbol{s}_0(\mathcal{X}) + \sum_i \boldsymbol{\theta}_i^T \boldsymbol{s}_i(\mathcal{Z}_i) + \tilde{\boldsymbol{\theta}}^T \tilde{\boldsymbol{s}}(\mathcal{Z}_i)\right)$$

The variational dual principle will tell us that the expected sufficient statistics:

$$\boldsymbol{\mu}_0^* = \langle \boldsymbol{s}_0 \rangle_P \qquad \boldsymbol{\mu}_i^* = \langle \boldsymbol{s}_i(\mathcal{Z}) \rangle_P \qquad \tilde{\boldsymbol{\mu}}_i^* = \langle \tilde{\boldsymbol{s}}_i \rangle_P$$

which are also the maximization of the likelihood, which is given by:

$$\{\boldsymbol{\mu}_0^*, \boldsymbol{\mu}_i^*, \tilde{\boldsymbol{\mu}}_i^*\} = \underset{\{\boldsymbol{\mu}_0, \boldsymbol{\mu}_i, \tilde{\boldsymbol{\mu}}_i\} \in \mathcal{M}}{\arg\max} \left[\boldsymbol{\theta}_0^T \boldsymbol{\mu}_0 + \sum_i (\boldsymbol{\theta}_i^T + \mathbf{0}^T \tilde{\boldsymbol{\mu}}_i) - \Phi(\boldsymbol{\mu}_0, \boldsymbol{\mu}_i, \tilde{\boldsymbol{\mu}}_i) \right]$$

Note that the negative entropy terms is evaluated over the full distribution i.e $\tilde{\boldsymbol{\theta}}$ doesn't have to be zero.

Remark 71. **(EP Relaxiation)** We consider the following relaxation to get the EP:

- We will have to relax the feasible sets $\mathcal{M}$, as we want a local consistency instead of global, where:
  - We want all edges connecting $\{\boldsymbol{\mu}_0, \tilde{\boldsymbol{\mu}}_i\}$, as $\boldsymbol{\mu}_0$ is the expected sufficient statistics over all $\mathcal{Z}$, thus we need it, when marginalized, to be consistent with all $\tilde{\mu}_i$
  - We want all pairs $(\boldsymbol{\mu}_i, \tilde{\boldsymbol{\mu}}_i)$ to be consistent also. This is equivalent to projection step (in EP) as we are matching expected sufficient statistics.
- The message entropy should be given by:

  $$\Phi_{\text{Bathe}}(\{\boldsymbol{\mu}_0, \tilde{\boldsymbol{\mu}}_i\}) - \sum_i (H[\boldsymbol{\mu}_i, \tilde{\boldsymbol{\mu}}_i] - H[\tilde{\boldsymbol{\mu}}_i])$$

  We have all graph structure entropy (without $\boldsymbol{\mu}_i$) together with conditional entropy (which is joint entropy minus by individual entropy)
- By doing this, we have dropped all the intractable terms and use the tractable terms to do the calculation (and talk to each other).
- Free-energy based approximation marginal include μ_i and run reparameterization on a junction graph.
- Direct learning on EP free-energy would use these marginal rather than approximate one and a local normalizer from the integration over $f_i(\mathcal{Z}_i) q_{\neg i}(\mathcal{Z}_i)$ (see the last part of the EP section.)

8 Variational Method

8.1 Introduction

Remark 72. **(Limitations)** Out treatment of variational method has emphasis natural choices of variational family often factorized using some functional (exponential) form as a joint. They are mostly restricted to joint exponential family facilitates hieratical and distributional model but not non-linear and non-conjugate.

Remark 73. **(Using Unconstraint Optimization)** Consider parameteric variational approximation via a constrained family $q(\mathcal{Z}; \boldsymbol{\rho})$. The constrained variational E-step becomes

$$q(\mathcal{Z}) = \underset{q \in \{q(\mathcal{Z}; \rho)\}}{\arg\max}\ \mathcal{F}(q(\mathcal{Z}), \boldsymbol{\theta}^{(k-1)}) \implies \boldsymbol{\rho}^{(k)} = \underset{\boldsymbol{\rho}}{\arg\max}\ \mathcal{F}(q(\mathcal{Z}; \boldsymbol{\rho}), \boldsymbol{\theta}^{(k-1)})$$

Remark 74. **(Reparameterized Free Energy)** We can replace constrained optimization $\mathcal{F}(q, \boldsymbol{\theta})$ with unconstrained optimization of constrained $\mathcal{F}(\rho, \boldsymbol{\theta})$:

$$\mathcal{F}(\rho, \boldsymbol{\theta}) = \left\langle \log P(\mathcal{X}, \mathcal{Z} | \boldsymbol{\theta}^{(k-1)}) \right\rangle_{q(\mathcal{Z}; \boldsymbol{\rho})} + H[\boldsymbol{\rho}]$$

We may use the coordinate ascent in $\boldsymbol{\rho}$ and $\boldsymbol{\theta}$ but this no longer necessary, as we have:

- In special case, the expectation of the log-joint under $q(\mathcal{Z}; \boldsymbol{\rho})$ can be expressed in closed form. We can follow $\nabla_{\boldsymbol{\rho}} \mathcal{F}$

- This requires evaluation a high-dimensional expectation with respected to $q(\mathcal{Z};\boldsymbol{\rho})$ as a function of $\boldsymbol{\rho}$ that isn't simple.
- There are 3 solutions to this problem:
  - "Score Based" gradient estimate and Monte-Carlow.
  - Recognition network training in separate place - not strictly variational.
  - Recognition network training simultaneously with generative model using frozen sample.

Proposition 8.1. *One can show that:*

$$\nabla_{\boldsymbol{\rho}}\mathcal{F}(\boldsymbol{\rho},\boldsymbol{\theta}) = \Big\langle [\nabla_{\boldsymbol{\rho}} \log q(\mathcal{Z};\boldsymbol{\rho})]\Big(\log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}) - q(\mathcal{Z};\boldsymbol{\rho})\Big)\Big\rangle_{q(\mathcal{Z};\boldsymbol{\rho})}$$

Proof. We consider the following gradient:

$$\begin{aligned}\nabla_{\boldsymbol{\rho}}\mathcal{F}(\boldsymbol{\rho},\boldsymbol{\theta}) &= \nabla_{\boldsymbol{\rho}} \int q(\mathcal{Z};)\Big[\log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}) - \log q(\mathcal{Z};\boldsymbol{\rho})\Big]\,\mathrm{d}\mathcal{Z} \\ &= \int [\nabla_{\boldsymbol{\rho}} q(\mathcal{Z};\boldsymbol{\rho})]\Big(\log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}) - \log q(\mathcal{Z};\boldsymbol{\rho})\Big) + q(\mathcal{Z};\boldsymbol{\rho})\nabla_{\boldsymbol{\rho}}\left[\log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}) - \log q(\mathcal{Z};\boldsymbol{\rho})\right]\,\mathrm{d}\mathcal{Z}\end{aligned}$$

We have the following facts:

$$\begin{aligned}&\nabla_{\boldsymbol{\rho}} \log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}) = 0 \\ &\int q(\mathcal{Z};\boldsymbol{\rho})\nabla_{\boldsymbol{\rho}} q(\mathcal{Z};\boldsymbol{\rho}) = \nabla_{\boldsymbol{\rho}} \int q(\mathcal{Z};\boldsymbol{\rho})\,\mathrm{d}\mathcal{Z} = 0 \\ &\nabla_{\boldsymbol{\rho}} q(\mathcal{Z};\boldsymbol{\rho}) = q(\mathcal{Z};\boldsymbol{\rho})\nabla_{\boldsymbol{\rho}} \log q(\mathcal{Z};\boldsymbol{\rho})\end{aligned}$$

And so we have the solution as required □

Remark 75. **(Reducing Variance)** To reduce the expectation of gradient, due to the high-dimensional problem, we can evaluate by MC.

- We can reduce by factorization, where $q(\mathcal{Z}) = \prod_i q(\mathcal{Z}_i|\boldsymbol{\rho}_i)$ factor over disjoint cliques.
- Let $\bar{\mathcal{Z}}_i$ be the minimal markov blancket of $\mathcal{Z}_i$ in the joint $P_{\bar{\mathcal{Z}}_i}$ be a product of joint factors that include element of $\mathcal{Z}_i$ and $P_{\neg\bar{\mathcal{Z}}_i}$

We have the following gradient:

$$\begin{aligned}\nabla_{\rho_i}\mathcal{F}(\{\rho_j\},\boldsymbol{\theta}) &= \left\langle \left[\nabla_{\rho_i}\sum_j \log q(\mathcal{Z}_j;\rho_j)\right]\left(\log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}) - \sum_j \log q(\mathcal{Z}_j;\rho_j)\right)\right\rangle_{q(\mathcal{Z})} \\ &= \left\langle [\nabla_{\rho_i}\log q(\mathcal{Z}_i;\boldsymbol{\rho}_i)](\log P_{\bar{\mathcal{Z}}_i}(\mathcal{X},\bar{\mathcal{Z}}_i) - \log q(\mathcal{Z}_i;\boldsymbol{\rho}_i))\right\rangle_{q(\bar{\mathcal{Z}}_i)} \\ &\quad + \left\langle [\nabla_{\rho_i}\log q(\mathcal{Z}_i;\boldsymbol{\rho}_i)]\left(\log P_{\neg\bar{\mathcal{Z}}_i}(\mathcal{X},\bar{\mathcal{Z}}_{\neg i}) - \sum_{j\neq i}\log q(\mathcal{Z}_j;\boldsymbol{\rho}_j)\right)\right\rangle_{q(\mathcal{Z})}\end{aligned}$$

Please note that the second term is propositional to $\langle\nabla_{\rho_i}\log q(\mathcal{Z}_i;\boldsymbol{\rho}_i)\rangle_{q(\mathcal{Z}_i)} = 0$ so we only need to consider the expectation with respected to $q(\bar{\mathcal{Z}}_i)$ which is variational message passing.

Remark 76. **(Sampling Methods)** We consider the following "black-box" variational approach is as follows:

- Choose a parameteric (factored) variational family $q(\mathcal{Z}) = \prod_i q(\mathcal{Z}_i;\boldsymbol{\theta})$
- Initiate the factors.

- Repeat until convergence:
  - Stochastic VE-step: Sample from $q(\bar{\mathcal{Z}}_i)$ and estimate expected gradient $\nabla\rho_i\mathcal{F}$, and we update ρ_i along the gradient.
  - Stochastic M-step: Sample from each $q(\bar{\mathcal{Z}}_i)$ as we can update the corresponding parameter.
- Stochastic may use Robbins Monro to promote convergence.
- Variance of the gradient estimate can also be controlled by MC techniques.

Remark 77. **(Batches of Data)** We have not distinguish between multi-variate models and iid data instances. As we group, all together in $\mathcal{Z}$; for example, large model such as HMM, we often make with multiple data draws and each instance requires a separate variational optimization.

Definition 8.1. (Recognition Model) Suppose we have fixed length vector $\{(\boldsymbol{x}_i, \boldsymbol{z}_i)\}$ when $\boldsymbol{z}_i$ is latent:

- Optimal variational distribution $q^*(\boldsymbol{z}_i)$ will depends on $\boldsymbol{x}_i$
- We want to learn the mapping from $q(\boldsymbol{z}_i; \boldsymbol{\rho} = f(\boldsymbol{x}_i; \boldsymbol{\phi}))$
- Now $\boldsymbol{\rho}$ is output of a general function approximate f parameterized by $\boldsymbol{\phi}$, training on map $\boldsymbol{x}_i$ to the variational parameter of $q(\boldsymbol{z}_i)$

The mapping function f is called recognition model.

Definition 8.2. (Helmholtz Model) It is a binary sigmoid belief-network with parallel recognition model. There are 2 phase of learning:

- *Wake Phase*: Given current f estimate mean field representation from data:

  $$q(z_i) = \text{Bern}(\hat{z}_i) \qquad \hat{z}_i = f(x_i; \boldsymbol{\phi})$$

  We will update generative parameter $\boldsymbol{\theta}$ according to $\nabla_{\boldsymbol{\theta}}\mathcal{F}(\{\hat{z}_i\}; \boldsymbol{\theta})$
- *Sleep Phase*: Sample $\{\boldsymbol{z}_i, \boldsymbol{x}_i\}_{i=1}^S$ from the current generative model. Update recognition parameter $f(\boldsymbol{x}_i)$ toward $\boldsymbol{z}_i$:

  $$\Delta\boldsymbol{\phi} \propto \sum_{i=1}^{S} (\boldsymbol{z}_i - f(\boldsymbol{x}_i; \boldsymbol{\phi}))\nabla_{\boldsymbol{\phi}} f(\boldsymbol{x}_i; \boldsymbol{\phi})$$

  Please note that this step minimizes:

  $$\text{KL}\left[P_\theta(\boldsymbol{z}|\boldsymbol{x}) \middle\| q(\boldsymbol{z}; f(\boldsymbol{x}; \boldsymbol{\phi}))\right]$$

  This is opposite to variational objective. Opposite to variational objective, but may not matter if divergence is small enough

Remark 78. **(Comments on Helmholtz Model Evaluation)** We have to sample $\boldsymbol{z}$ from recognition model rather than just evaluate means:

- Expectation in free-energy can be computed directly rather than by means substuition.
- In hieratical model, output of higher recognition layer depends on sample at previous stages, which introduces correlation between sample at difference layer.

Recognition model structure need not necessary exactly echo generative model. Please note that a more general approach is to train f to yields the parameter of exponential family $q(\boldsymbol{z})$.

Definition 8.3. (Variational Autoencoder) We fuses the wake and sleep phase. We generate the recognition sample used deterministic transformation of external random variable. If f gives marginal μ_i and σ_i for latent z_i and $\boldsymbol{\varepsilon}_i^s \sim \mathcal{N}(0,1)$ then:

$$\boldsymbol{z}_i^s = \boldsymbol{\mu}_i + \sigma_i \boldsymbol{\varepsilon}_i^s$$

Given the generative and recognition model can be trained together with gradient descent. Holding $\boldsymbol{\varepsilon}^s$ fixed:

$$\mathcal{F}_i(\boldsymbol{\theta}, \boldsymbol{\phi}) = \sum_s \log P(\boldsymbol{x}_i, \boldsymbol{z}_i^s; \boldsymbol{\theta}) - \log q(\boldsymbol{z}_i^s; f(\boldsymbol{x}_i, \boldsymbol{\phi}))$$

We have the following derivative as:

$$\frac{\partial}{\partial \boldsymbol{\theta}} \mathcal{F}_i = \sum_s \nabla_{\boldsymbol{\theta}} \log P(\boldsymbol{x}_i, \boldsymbol{z}_i^s; \boldsymbol{\theta})$$

$$\frac{\partial}{\partial \boldsymbol{\phi}} \mathcal{F}_i = \sum_s \frac{\partial}{\partial z_i^s} \Big(\log P(\boldsymbol{x}_i, \boldsymbol{z}_i^s; \boldsymbol{\theta}) - \log q(\boldsymbol{z}_i^s; f(\boldsymbol{x}_i)) \Big) \frac{d\boldsymbol{z}_i^s}{d\phi} + \frac{\partial}{\partial f(\boldsymbol{x}_i)} \log q(\boldsymbol{z}_i^s; f(\boldsymbol{x}_i)) \frac{df(\boldsymbol{x}_i)}{d\phi}$$

Remark 79. **(Observations on Variational Auto-Encoder)** We consider the following observations on the training of VAE:

- We start by frozen sample $\boldsymbol{\varepsilon}^s$ can be redrawn that avoid overfitting.
- It maybe possible to evaluate the entropy and $\log P(\boldsymbol{z})$ without sampling and reducing variance:
- Differentiable reparameterization are avaliable for number of difference distribution
- The conditional $P(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\theta})$ is often implemented as a neural network with additive noise at input.
- In practice, hieratical model appear to be difficult to train.

8.2 Additional Models to VAE

Definition 8.4. (Importance Weigh Free Energy) Consider another interpretation of free energy:

$$\mathcal{F}(q; \boldsymbol{\theta}) = \left\langle \log \frac{P(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} \right\rangle_q$$

We consider the jensen's inequality on importance sampled estimate:

$$l(\boldsymbol{\theta}) = \log \mathbb{E}_{z \sim q} \left[\frac{P(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} \right] \leq \mathbb{E}_{z \sim q} \left[\log \frac{P(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} \right] = \mathcal{F}(q; \boldsymbol{\theta})$$

Suggest more accurate importance weight as we have:

$$l(\boldsymbol{\theta}) = \log \mathbb{E}_{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_k \sim q} \left[\frac{1}{k} \sum_k \frac{P(\boldsymbol{x}_k, \boldsymbol{z}_k)}{q(\boldsymbol{z}_k)} \right] \geq \mathbb{E}_{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_k \sim q} \left[\log \frac{1}{k} \sum_k \frac{P(\boldsymbol{x}_k, \boldsymbol{z}_k)}{q(\boldsymbol{z}_k)} \right]$$

This allows the tight-bound and reparameterization friendly but as $K \to \infty$, the signal for learning amortized q grows weaker making VAE learning too slow.

Definition 8.5. (Normalizing Flow) We have the following free energy:

$$\mathcal{F}(q, \boldsymbol{\theta}) = \langle \log P(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}) \rangle_q - \langle \log q(\boldsymbol{z}) \rangle_q$$

To evaluate $\mathcal{F}$, we need to be able to find expectation with respected to q and evaluate the log-density, which usually restrict us to tractable inference families. We consider the followign recognition model $q(\boldsymbol{z})$ implicitly by:

$$\boldsymbol{z}_0 \sim q(\cdot; \boldsymbol{x}) \qquad \boldsymbol{z} = f_k(f_{k-1}(\cdots f_1(\boldsymbol{z}_0)))$$

where q_0 should be fixed and tractable. And so, we have the following evaluations:

$$\langle F(\boldsymbol{z})\rangle_q = \langle F(f_k(f_{k-1}(\cdots f_1(\boldsymbol{z}_0)))))\rangle_{q_0} \qquad \log q(\boldsymbol{z}) = \log q_0(f_1^{-1}(\cdots f_{k-1}^{-1}(f_k^{-1}(\boldsymbol{z})))) - \sum_k \log|\nabla f_k|$$

where $|\nabla f_k|$ being the determinant of the jacobian, where we use the following transformation of variables:

$$\boldsymbol{z}_k = f_k(\boldsymbol{z}_{k-1}) \quad \implies \quad q(\boldsymbol{z}_k) = q(f_k^{-1}(\boldsymbol{z}_k))\left|\frac{\partial \boldsymbol{z}_{k-1}}{\partial \boldsymbol{z}_k}\right| = q(f_k^{-1}(\boldsymbol{z}_k))\,|\nabla f_k(\boldsymbol{z}_{k-1})|^{-1}$$

Given a sample $\boldsymbol{z}_0^i \sim q_0(\cdot;\boldsymbol{x})$ as we have:

$$\mathcal{F}(q,\boldsymbol{\theta}) \approx \frac{1}{S}\sum_s \log p(\boldsymbol{x}, f_k(f_{k-1}(\cdots f_1(\boldsymbol{z}_0^i)))) + h[q_0] + \frac{1}{S}\sum_s\sum_k \left|\nabla f_k(f_{k-1}(\cdots f_1(\boldsymbol{z}_0^i)))\right|$$

Remark 80. **(Special f for Normalizing Flow)** Suppose we use the special f and we have:

$$f(\boldsymbol{z}) = \boldsymbol{z} + \boldsymbol{u}h(\boldsymbol{w}^T\boldsymbol{z}+b) \implies |\nabla f| = \left|1+\boldsymbol{u}^T\boldsymbol{\Psi}(\boldsymbol{z})\right| \quad \text{where} \quad \Psi(\boldsymbol{z}) = h'(\boldsymbol{x}^T\boldsymbol{z}+b)\boldsymbol{w}$$

$$f(\boldsymbol{z}) = \boldsymbol{z} + \frac{\beta}{\alpha+|\boldsymbol{z}-\boldsymbol{z}_0|} \implies |\nabla f| = [1+\beta h]^{d-1}\,[1+\beta h+\beta h' r] \quad \text{where} \quad r = |\boldsymbol{z}-\boldsymbol{z}_0| \quad h = \frac{1}{\alpha+r}$$

Definition 8.6. (DDC Helmholz Machine) We define q to be unnormalized exponential family with *large* set of sufficient statistics:

$$q(\boldsymbol{z}) \propto \exp\left(\sim_i \eta_i\psi_i(\boldsymbol{z})\right)$$

and it is parameterized by mean parameter $\boldsymbol{\mu} = \langle\boldsymbol{\psi}(\boldsymbol{z})\rangle$, which we call distributed distribution code (DDC). Train recognition model using a sleep sample as we have:

$$\boldsymbol{\mu} = \langle\boldsymbol{\psi}(\boldsymbol{z})\rangle_q = f(\boldsymbol{x}^*;\boldsymbol{\phi})$$
$$\Delta\boldsymbol{\phi} \propto \sum_s (\boldsymbol{\psi}(\boldsymbol{z}_s) - f(\boldsymbol{x}_s;\boldsymbol{\phi}))\nabla_{\boldsymbol{\phi}} f(\boldsymbol{x}_s;\boldsymbol{\phi})$$

Furthermore, we also learn linear approximation $\nabla\log P(\boldsymbol{x},\boldsymbol{z}|\boldsymbol{\theta}) \approx \boldsymbol{A}\psi(\boldsymbol{z})$, where

$$\boldsymbol{A} = \left(\sum_s \nabla\log P(\boldsymbol{x}_s,\boldsymbol{z}_s|\boldsymbol{\theta})\psi(\boldsymbol{z}_s)\right)^T\left(\sum_s \psi(\boldsymbol{z}_s)\psi(\boldsymbol{z}_s)^T\right)^{-1}$$

Then we have $\langle\nabla\log P(\boldsymbol{x},\boldsymbol{z})\rangle_q \approx \boldsymbol{A}\langle\boldsymbol{\psi}(\boldsymbol{z})\rangle \propto f(\boldsymbol{x},\boldsymbol{\phi})$ can be generalized into infinite dimension with kernel.

Definition 8.7. (Amortised Learning) We aren't interested in inference. We can short-circled general recognition and compute expectation for learning directly, as we have:

$$\nabla_{\boldsymbol{\theta}} l(\boldsymbol{\theta}) = \frac{\partial}{\partial\boldsymbol{\theta}}\mathcal{F}(q^*,\boldsymbol{\theta}) = \langle\nabla_{\boldsymbol{\theta}}\log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta})\rangle_{q^*}$$

We can use the wake-sleep approach:

- Sample $\{\boldsymbol{x}_s,\boldsymbol{z}_s\} \sim P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta}^k)$
- Train Regression $\hat{J}_{\boldsymbol{\theta}^k} : \boldsymbol{x}_k \mapsto \nabla_{\boldsymbol{\theta}}\log P(\boldsymbol{x}_s,\boldsymbol{z}_s)|_{\boldsymbol{\theta}^k}$ (Learning the mapping)
- Set $\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + \eta\sum_i \hat{J}_{\boldsymbol{\theta}^k}(\boldsymbol{x}_i)$

Derivate form works for (kernel and GP) regression for which regressor is linear in target. For conditional, exponential family model:

$$\langle\log P(\mathcal{X},\mathcal{Z}|\boldsymbol{\theta})\rangle_{q^*} = \langle\boldsymbol{\eta}(\boldsymbol{z},\boldsymbol{\theta})\rangle_{q^*}^T\boldsymbol{T}(\boldsymbol{x}) - \langle\boldsymbol{\Phi}(\boldsymbol{z},\boldsymbol{\theta}) + \log P(\boldsymbol{z}|\boldsymbol{\theta})\rangle_{q^*}$$

and regressor can be trained to function of $\boldsymbol{z}$ alone, with $T(\boldsymbol{x})$ evaluated on (wake-phase) data.

Remark 81. **(VAE Comments)** Much of the VAE and related work has common generative model as:

$$\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \qquad \boldsymbol{x} \sim \mathcal{N}(\boldsymbol{g}(\boldsymbol{z}; \boldsymbol{\theta}), \psi \boldsymbol{I})$$

where $\boldsymbol{g}$ is neural network. Let's consider the dimension of $\boldsymbol{z}$ as we have:

- Overcomplicated: If $\dim(\boldsymbol{z})$ is large enough, the optimal solution has $\psi \to 0$ as $\boldsymbol{q}(\boldsymbol{z}; \boldsymbol{x}) \to \delta(\boldsymbol{z} - \boldsymbol{f}(\boldsymbol{x}; \boldsymbol{\phi}))$. In effect, the generative model learns a flow to transform a model density to the target.
- Oversimplified: If $\dim(\boldsymbol{z})$ is small as non-linear PCA.

Interesting latent representation are required more structured generative model.

Definition 8.8. (Structured VAE) Consider a model where $P(\mathcal{Z}|\boldsymbol{\theta})$ has tractable joint exponential family potential and:

$$P(\mathcal{X}|\mathcal{Z}, \Gamma) = \prod_i P(\boldsymbol{x}_i|\boldsymbol{z}_i, \boldsymbol{\gamma}_i)$$

are interactable. conditional independent observation $\boldsymbol{\gamma}_i$ might be the same for all i. Consider factored variational inference $q(\mathcal{Z}) = \prod_i q(\boldsymbol{z}_i)$ with no further constriants:

$$\begin{aligned}
\log q_i^*(\boldsymbol{z}_i) &= \langle \log P(\mathcal{Z}, \mathcal{X}) \rangle_{q_{\neg i}} + \text{const.} \\
&= \langle \log P(\boldsymbol{z}_i|\mathcal{Z}_i) + \log P(\boldsymbol{x}_i|\boldsymbol{z}_i) \rangle_{q_{\neg i}} + \text{const.} \\
&= \langle \boldsymbol{\eta}_{\neg i} \rangle_{q_{\neg i}}^T \boldsymbol{\psi}_i(\boldsymbol{z}_i) + \log P(\boldsymbol{x}_i|\boldsymbol{z}_i)
\end{aligned}$$

Let's consider each variables (exploited the exponential family form of $P(\mathcal{Z})$):

- $\boldsymbol{\psi}_i$ are effective sufficient statistics included log-normalizer of children of DAG.
- $\boldsymbol{\eta}_{\neg i}$ is function of $\mathcal{Z}_{\neg i}$

We will choose the parameteric form of $q_i(\boldsymbol{z}_i) = \exp(\tilde{\boldsymbol{\eta}}^T \boldsymbol{\psi}_i(\boldsymbol{z}_i) - \boldsymbol{\Phi}_i(\tilde{\boldsymbol{\eta}}_i))$ and so the optimum will have:

$$\log q_i^*(\boldsymbol{z}_i) = \langle \boldsymbol{\eta}_{\neg i} \rangle_{q_{\neg i}}^T \boldsymbol{\psi}_i(\boldsymbol{z}_i) + \boldsymbol{\rho}(\boldsymbol{x}_i)^T \boldsymbol{\psi}_i(\boldsymbol{z}_i)$$

where $\boldsymbol{\rho}(\boldsymbol{x}_i) = \boldsymbol{f}_i(\boldsymbol{x}_i; \boldsymbol{p}hi)$ recognition function and it might be the same for all i.

Remark 82. **(Training of Structured VAE)** We consider the free energy:

$$\begin{aligned}
\mathcal{F}(\boldsymbol{\theta}, \boldsymbol{\Gamma}, \{\boldsymbol{\phi}_i\}) &= \left\langle \sum_i \log P(\boldsymbol{x}_i|\boldsymbol{z}_i, \boldsymbol{\gamma}_i) + \log P(\mathcal{Z}|\boldsymbol{\theta}) \right\rangle_{q(\mathcal{Z}; \boldsymbol{\theta}, \langle \boldsymbol{\phi} \rangle)} + \sum_i H[q_i] \\
&= \sum_i \underbrace{\langle \log P(\boldsymbol{x}_i|\boldsymbol{z}_i, \boldsymbol{\gamma}_i) + \log P(\mathcal{Z}|\boldsymbol{\theta}) \rangle_{q_i(\boldsymbol{z}_i; \boldsymbol{\theta}, \boldsymbol{\phi}_i)} + H[q_i]}_{\mathcal{F}_i} + \langle \log P(\mathcal{Z}|\theta) \rangle_{q(\mathcal{Z}, \boldsymbol{\theta}, \{\boldsymbol{\phi}_i\})}
\end{aligned}$$

Update $\boldsymbol{\theta}$ are just tractable model. To update each ϕ_i and γ_i, find $\langle \boldsymbol{\eta}_{\neg i} \rangle_{q_{\neg i}}$ to give the prior-like in VAE, and we generated the reparameterization sample $\boldsymbol{z}_i^s \sim q_i$, then:

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\gamma}_i} \mathcal{F}_i &= \sum_s \nabla_{\boldsymbol{\gamma}_i} \log P(\boldsymbol{x}_i, \boldsymbol{z}_i^s; \boldsymbol{\gamma}_i) \\
\frac{\partial}{\partial \boldsymbol{\phi}_i} \mathcal{F}_i &= \sum_s \frac{\partial}{\partial \boldsymbol{z}_i^s} \Big(\log P(\boldsymbol{x}_i, \boldsymbol{z}_i^s; \boldsymbol{\gamma}_i) - \log q(\boldsymbol{z}_i^s; f(\boldsymbol{x}_i)) \Big) \frac{d\boldsymbol{z}_i^s}{d\boldsymbol{\phi}} + \frac{\partial}{\partial f(\boldsymbol{x}_i)} \log q(\boldsymbol{z}_i^s; f(\boldsymbol{x}_i)) \frac{df(\boldsymbol{x}_i)}{d\boldsymbol{\phi}}
\end{aligned}$$

This is like standard VAE.