# Algorithms and solutions to exercises

Ivano Lodato

February 8, 2024

The rules of do-calculus are equalities valid (and applicable in both directions) whenever certain conditional independecies exist in a mutilated version of the causal graph. The whole point of conditioning is to force independencies between variables. In short, d-separation on a graph, or conditional independence in probability distributions, is the name of the game.

Say we have an interventional query, $P(Y|\hat{X}, Z)$, where $Y, X, Z$ are sets of nodes/variables. The set $Z$ may be empty, but the set $Y$ and $X$ must contain some elements, otherwise we would have no outcome $Y$ (the probability distribution is defined over nothing) or intervention(s) $X$ (the query would be non-interventional, hence trivial). The procedure to solve a query of this kind on the graph $G$ is explained below, with images and counterexamples.

## Solve interventional queries - single outcome

Start by fixing one of the interventions $x \in X$. It can be proven easily that the final result will not depend on the order of interventions.

1. Find *all* the active paths between an intervention $x \in X$ and the outcome $y \in Y$. Call the set of these paths $\mathcal{P}$
   If a path from $x \in X$ to $y \in Y$ is inactive, it either means the flow of information is blocked by an observation of a fork or chain node $z \in Z$, or an unobserved collider $c$ ($c \notin Z$)

2. Divide $\mathcal{P}$ into backdoor ($\mathcal{P}_B$) and frontdoor ($\mathcal{P}_F$) paths

   (a) If $\mathcal{P}_B = \emptyset$ and $\mathcal{P}_F = \emptyset$, try to use rule 3 of do-calculus (next steps to be added)

   (b) If the set $\mathcal{P}_B = \emptyset$ is empty (there is no active backdoor path), and $\mathcal{P}_F \neq \emptyset$, use rule 2 of do-calculus

   (c) If the set $\mathcal{P}_B \neq \emptyset$ is empty (there is no active backdoor path), and $\mathcal{P}_F = \emptyset$, try to use rule 3 of do-calculus (next steps to be added)

   (d) $\mathcal{P}_B \neq \emptyset$ and $\mathcal{P}_F \neq \emptyset$, conditioning (observing) some variable is necessary.
   If we have both backdoor and frontdoor paths active, then we cannot use either rule 2 or rule 3 (or no sequence of both) of do-calculus to identify the interventional query.

   Check among the active paths which are *unblockable*: i.e. (1) conditioning (i.e. observing a fork or a chain) on that active path is not possible, because either there are no fork/chain nodes on that path (paths of length 1) or they are unobserved; or (2) a collider on a path is observed, and there are no fork/chain nodes on that path (or they are unobserved) that we can condition the probability distribution on).
   Call the sets of unblockable paths $\mathcal{P}_B^U$ (backdoor) and $\mathcal{P}_F^U$ (frontdoor)

   i. if there are no unblockable backdoor paths $\mathcal{P}_B^U = \emptyset$, then we condition on fork/chain nodes on those paths and block them (see below *). We are left with some active frontdoor paths (unblockable and blockable): use rule 2 of do-calculus

ii. if there are no unblockable frontdoor paths, $\mathcal{P}_F^U = \emptyset$ then we condition on fork/chain nodes on those paths, and block them (see below *). We are left with some active backdoor paths (unblockable and blockable): try to use rule 3 of do-calculus (next steps to be added)

iii. if there are no unblockable (backdoor or frontdoor) paths, then decide which paths you want to block (see below *), and use the corresponding rule of do-calculus to (try to) simplify the intervention $x$

iv. if there is at least one frontdoor and one backdoor unblockable paths, then the query is **non-identifiable**

Repeat this procedure for all interventions in all probability distributions of the query (remember, after conditioning the joint probability distributions is expanded and new probability distributions appear with interventions as conditions) until they are either transformed into observations (via rule 2) or erased from the query (via rule 3). Finally, check if rule 1 can be applied on any observations - as the straightforward application of rule 2 (whenever only front-door paths are active) may lead to an superfluous observation, i.e. we could have deleted the intervention via rule 3 instead of using rule 2 and then rule 1 on it.
*If even one intervention in one probability distribution cannot be transformed or deleted after conditioning, then the graph is non-identifiable.*
[**Exercise 3: Check and identify, in the solutions of Exercise 1 below, which intervention was deleted via rule 3 and which observation was deleted via rule 1. Pay attention to how and when rule 3 is used.**].

[*** finding the correct variables to condition on is not trivial. Conditioning on a variable $z$ on a path between $x$ and $y$ to block it, may activate some other path on which that variable was a collider. So now we may have to block that path (See worked out example below.)**]

# 1 Solve interventional queries - many outcomes

The above procedure can be generalized, by still considering one intervention $x \in X$ at a time, but there are now many outcomes $y_1, \dots, y_n \in Y$. Step 1 of the algorithm above gets modified into:

① Find *all* active paths between $x$ and all the outcomes in $Y$. Call the paths from $x$ to each outcome $y_i$ $\mathcal{P}^i$.

② Divide each of these path sets into backdoor $\mathcal{P}_B^i$ and frontdoor $\mathcal{P}_F^i$ active paths.

- ⓐ If we have $\mathcal{P}_B^i = \emptyset$ and $\mathcal{P}_F^i = \emptyset$ for all outcomes $y_i$, then we try to use rule 3 directly on $x$
- ⓑ If we have $\mathcal{P}_B^i = \emptyset$ but $\mathcal{P}_F^i \neq \emptyset$, then we can use rule 2 on $x$
- ⓒ If we have $\mathcal{P}_F^i = \emptyset$ but $\mathcal{P}_B^i \neq \emptyset$, then try to use rule 3 on $x$
- ⓓ If $\mathcal{P}_F^i \neq \emptyset$ and $\mathcal{P}_B^i \neq \emptyset$, conditioning on some variable is necessary.
  Check, for each $\mathcal{P}^i$ which are the unblockable paths. Call them $P_B^{i,U}$ and $P_F^{i,U}$. For each outcome $y_i$, any of these two sets may be empty, so there are many possible such combinations. Let us go in order:
  - ⓘ Suppose there is a specific outcome $y_i$ for which exist at least one unblockable active backdoor and one unblockable active backdoor. In this case, we can already say that the query is non-identifiable
  - ⓘⓘ Suppose for no outcome $y_i \in Y$, there exist both an unblockable backdoor and frontdoor paths from $x$. This implies that, for each outcome $y_i$ there may be at most one type of paths which are unblockable, either $\mathcal{P}_B^{i,U} = \emptyset$ or $\mathcal{P}_F^{i,U} = \emptyset$. If for all outcomes, only

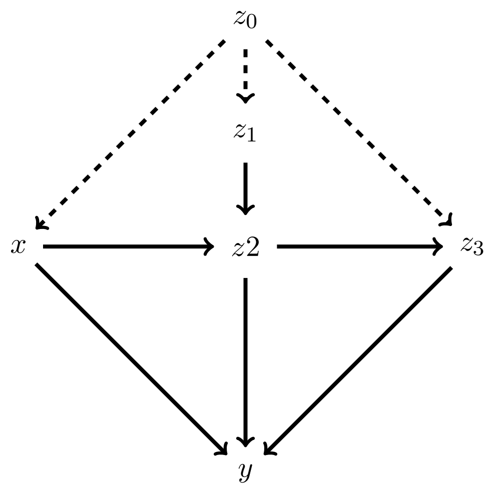



Figure 1: (simplified) Causal graph presented in Figure 3.1 of the book "Causality: models,reasoning, and inference"

unblockable frontdoor paths exist, we condition on forks/chain on the backdoor paths from $x$ to all outcomes $y_i$, then apply rule 2 on the intervention

(iii) If for all outcomes, only unblockable backdoor paths exist, we condition on forks/chain on the frontdoor paths from $x$ to all outcomes $y_i$, then try to apply rule 3 on the intervention

(iv) If for $k$ outcomes the unblockable paths are frontdoor, and for $k - n$ outcomes the unblockable paths are backdoor, we find all variables which block as many of the active backdoor paths connecting to the $k$ outcomes and, at the same time, as many of the active frontdoor paths connecting to the $n - k$ outcomes (pay attention to the observation * above). Eventually, we will need to decompose the full joint probability distribution $P(y_1, \dots, y_n|\hat{x})$ into the product of many smaller joint probability distributions, as dictated by the blocking variables in the blocked paths. Then start from step 1 again.

Let us work out an exercise together. We try to answer the query $P(y|\hat{x})$ int he graph of Figure 1. We use the steps for one outcome $y$. First we identify (in)active paths between $x$ and $y$.

Active frontdoor paths

$$x \to y \quad ; \tag{1}$$

$$x \to z_2 \to y \quad ; \tag{2}$$

$$x \to z_2 \to z_3 \to y \quad ; \tag{3}$$

Inactive frontdoor paths

$$x \to z_2 \leftarrow z_1 \leftarrow z_0 \to z_3 \to y \quad ; \tag{4}$$

Active backdoor paths

$$x \leftarrow z_0 \to z_3 \to y \quad ; \tag{5}$$

$$x \leftarrow z_0 \to z_1 \to z_2 \to y \quad ; \tag{6}$$

$$x \leftarrow z_0 \to z_1 \to z_2 \to z_3 \to y \quad ; \tag{7}$$

Inactive backdoor paths

$$x \leftarrow z_0 \to z_3 \leftarrow z_2 \to y \quad ; \tag{8}$$

Among the active frontdoor paths, one is unblockable $x \to y$. The other two paths can be blocked by ($z_2$) and ($z_2$ and/or $z_3$) respectively. Note also that observing $z_2$ activates the inactive frontdoor path ($z_2$ is a collider there), but observing also $z_3$ blocks (keep inactive) this path.
Among the backdoor paths, none is unblockable: the active backdoor paths can be blocked respectively by ($z_3$), ($z_1$ and/or $z_2$), ($z_1$ and/or $z_2$ and/or $z_3$). Note that, again, observing $z_3$ activates an inactive frontdoor path, but observing also $z_2$ keeps the path inactive. Hence, by looking at the intersection of the sets of variables blocking the backdoor paths, we immediately obtain ($z_2$,$z_3$) ( $z_3$ is the only variable that can block the path $x \leftarrow z_0 \to z_3 \to y$, once $z_3$ is conditioned on, then we

need to condition on $z_2$ to keep the backdoor path $x \leftarrow z_0 \rightarrow z_3 \leftarrow z_3 \rightarrow y$ inactive. Finally, we note that all other backdoor paths are blocked by $(z_2, z_3)$, and no other spurious path is activated by these variables.
The conditioning goes as follows:

$$P(y|\hat{x}) = \sum_{z_2,z_3} P(y|\hat{x}, z_2, z_3)P(z_2, z_3|\hat{x}) = \sum_{z_2,z_3} P(y|x, z_2, z_3)P(z_2, z_3|\hat{x}) \tag{9}$$

where we applied rule 2 on the first probability distribution in the second equality. We have another probability distribution to analyse, $P(z_2, z_3|\hat{x})$ (see the steps to follow for interventional queries with many outcomes). Again, we find all active paths from $x$ to both $z_2$ and $z_3$.

Active frontdoor paths $x \rightarrow \dots z_2$

$$x \rightarrow z_2 \quad ; \tag{10}$$

Inactive frontdoor paths $x \rightarrow \dots z_2$

$$x \rightarrow y \leftarrow z_2 \quad ; \tag{11}$$

$$x \rightarrow y \leftarrow z_3 \leftarrow z_2 \quad ; \tag{12}$$

$$x \rightarrow y \leftarrow z_3 \leftarrow z_0 \rightarrow z_1 \rightarrow z_2 \quad ; \tag{13}$$

Active backdoor paths $x \leftarrow \dots z_2$

$$x \leftarrow z_0 \rightarrow z_1 \rightarrow z_2 \quad ; \tag{14}$$

Inactive backdoor paths $x \leftarrow \dots z_2$

$$x \leftarrow z_0 \rightarrow z_3 \leftarrow z_2 \quad ; \tag{15}$$

$$x \leftarrow z_0 \rightarrow z_3 \rightarrow y \leftarrow z_2 \quad ; \tag{16}$$

Active frontdoor paths $x \rightarrow \dots z_3$

$$x \rightarrow z_2 \rightarrow z_3 \quad ; \tag{17}$$

Inactive frontdoor paths $x \rightarrow \dots z_3$

$$x \rightarrow z_2 \leftarrow z_1 \leftarrow z_0 \rightarrow z_3 \quad ; \tag{18}$$

$$x \rightarrow y \leftarrow z_3 \quad ; \tag{19}$$

$$x \rightarrow z_2 \rightarrow y \leftarrow z_3 \quad ; \tag{20}$$

$$x \rightarrow y \leftarrow z_2 \rightarrow z_3 \quad ; \tag{21}$$

$$x \rightarrow y \leftarrow z_2 \leftarrow z_1 \leftarrow z_0 \rightarrow z_3 \tag{22}$$

Active backdoor paths $x \leftarrow \dots z_3$ ;

$$x \leftarrow z_0 \rightarrow z_3 \quad ; \tag{23}$$

$$x \leftarrow z_0 \rightarrow z_1 \rightarrow z_2 \rightarrow z_3 \quad ; \tag{24}$$

Inactive backdoor paths $x \leftarrow \dots z_3$

$$x \leftarrow z_0 \rightarrow z_1 \rightarrow z_2 \rightarrow y \leftarrow z_3 \quad ; \tag{25}$$

Now there is an unblockable frontdoor path from $x \rightarrow z_2$. The active backdoor path to $z_2$ can be blocked by conditioning on $z_1$ (and this will not activate any other paths). On the other hand, there is an unblockable backdoor path to $z_3$, $x \leftarrow z_0 \rightarrow z_3$. The directed frontdoor path $x \rightarrow z_2 \rightarrow z_3$ can be blocked only by conditioning on $z_2$. However, this activates the path $x \rightarrow z_2 \leftarrow z_1 \leftarrow z_0 \rightarrow z_3$ which we can block only by observing $z_1$. So conditioning the joint probability distribution on $z_1$ and $z_2$, we block at the same time the active backdoor path to $z_2$ $x \leftarrow z_0 \rightarrow z_1 \rightarrow z_2$, the frontdoor path $x \rightarrow z_2 \rightarrow z_3$ (and frontdoor path, activated by observing $z_2$ $x \rightarrow z_2 \leftarrow z_1 \leftarrow z_0 \rightarrow z_3$). It is also to be noticed that one of the blocking variable $z_2$, in this case, is also an outcome. So *we first condition on all blocking variables available which are not outcomes*, i.e. $z_1$:

$$P(z_2, z_3|\hat{x}) = \sum_{z_1} P(z_2, z_3|\hat{x}, z_1)P(z_1|\hat{x}) \tag{26}$$

Now we note that the probability $P(z_2|w, z_2) = P(z_2|w)$. This is a trivial identity, since observing $z_2$ just fixes the value of the outcome, already fixed. So adding as an observation the outcome variable, doesn't change anything in the probability distribution (This about this!!!). So we can re-write the first factor above as:

$$P(z_2, z_3|\hat{x}, z_1) = \sum_{z_1} P(z_2, z_3|\hat{x}, z_2, z_1) \tag{27}$$

Now, since we have an unblockable backdoor path from $x$ to $z_3$ and an unblockable frontdoor path from $x$ to $z_2$, we cannot solve the intervention on the joint probability distributions by using any rules of do-calculus (or sequence thereof). Instead, we factorize the join probability distribution. Now, there is $n!$ way to factorize a joint probability distribution with $n$ outcomes. In this case, we see immediately that $z_2$ is needed as an observation to block the front-door path from $x$ to $z_3$, while $z_3$ is not needed to block any active paths from $x$ to $z_2$. So we can write:

$$\begin{aligned}\sum_{z_1} P(z_2, z_3|\hat{x}, z_1, z_2)P(z_1|\hat{x}) &= \sum_{z_1} P(z_3|\hat{x}, z_2, z_1)P(z_2|\hat{x}, z_1, z_2)P(z_1|\hat{x}) \\ &= \sum_{z_1} P(z_3|\hat{x}, z_2, z_1)P(z_2|\hat{x}, z_1)P(z_1|\hat{x}) \end{aligned} \tag{28}$$

where we have used again the identity $P(z_2|\hat{x}, z_1, z_2) = P(z_2|\hat{x}, z_1)$.
Clearly, the ordering can be attributed to the ancestral relation between $z_2$ and $z_3$.

The above procedure can be inverted, i.e. instead of conditioning on $z_1$ and observing (redundantly) $z_2$ and then factorizing the joint probability distribution, we can do the opposite: first we factorize the joint probability distribution, then condition on the variables needed. Now, following the ancestry relation of the variables $z_2, z_3$, it is easy to write:

$$P(z_2, z_3|\hat{x}) = P(z_3|\hat{x}, z_2)P(z_2|\hat{x}) \tag{29}$$

To identify the first factor, we need to note that the observation of $z_2$ activates a frontdoor path to $z_3$, which can be blocked by conditioning on $z_1$. There is again an unblockable backdoor path. We can then write two possible conditioning:

$$P(z_3|\hat{x}, z_2)P(z_2|\hat{x}) = \sum_{z_1} P(z_3|\hat{x}, z_2, z_1)P(z_1|\hat{x}, z_2)P(z_2|\hat{x}) \tag{30}$$

or alternatively:

$$P(z_3|\hat{x}, z_2)P(z_2|\hat{x}) = \sum_{z_1} P(z_3|\hat{x}, z_2, z_1)P(z_2|\hat{x}, z_1)P(z_1|\hat{x}) \tag{31}$$

The choice is not arbitrary: the first identity is wrong, because we are not accounting for the fact that $z_1$ is also needed to block paths from $x$ to $z_2$, not just paths from $x$ to $z_3$. Hence, we need to condition both $P(z_3|..)$ and $P(z_2|...)$ with $z_1$. Another way to see this is to consider the ancestry relations of all the outcomes that follow $z_3$ (including $z_3$ itself): in this case, we are manipulating the probability distribution with outcome $z_3$, so we need to order $z_3$,$z_2$ and $z_1$ (the last conditioning variable, for now) based on their ancestry.
It is clear that this alternative procedure gives equivalent results: however, the interpretation can become ambiguous and it complicates much in the way of generalizations, so we will stick with the first resolution: when considering a query with more than 1 outcome, we condition on the variables which are needed to block, at the same time, paths from $x$ to every outcome, paying attention to conditioning variables which are outcomes in the joint probability distribution. Then we decompose the joint probability distribution in factors guided by the conditioning variables needed (i.e. ancestry relations), as we have done in the example above.
**[Exercise 4: Complete the identification of the query $P(y|\hat{x})$ from the above results. Obtain the formula presented in eq. 3.1 of the book "Causality"]**

## Solution to Exercise 0

Every intervention (yellow node) $\hat{x}$ in the query is connected to other nodes in the graphs *only* through front door paths, so we can always use rule 2 of do-calculus. Note however, that in some cases, we can simplify the query further, by using rule 1 of do-calculus on an observation $x$, after using rule 2 on the intervention $\hat{x}$. Take for instance the case (it is in fact a subgraph in the picture) in which $x \rightarrow z \leftarrow y$. If $z$ is not observed, this path is inactive, so $x$ is d-separated from $y$. Furthermore, if the query is $P(y|\hat{x})$ (hence $z$, descendant of $x$ is not observed), I can directly use rule 3 of do-calculus and get rid of the intervention. An even simpler example is the case where no (active or inactive) path exist between $x$ and an outcome $y$: in this case, trivially, we can use rule 3.

Note 1: if the query on the above graph would have been $P(y|\hat{x}, z)$ then the path is active and unblockable! We can still use rule 2 of do-calculus, and transform the intervention $\hat{x} \to x$, but we cannot use rule 3 anymore!!!!

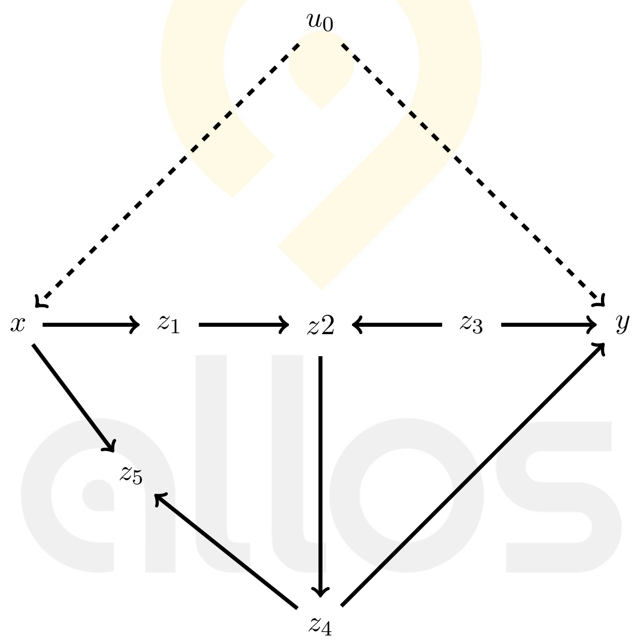
Note 2: if a variable $x$ satisfies rule 1 of do-calculus, $\hat{x}$ automatically satisfies rule 2 and rule 3. The viceversa is not always true.

In the above example, for instance, with the graph $x \to z \leftarrow y$ and $z$ is not observed, we can apply rule 3 or apply rule 2 and then rule 1, but this is not always the case.

**[Exercise 5: Draw a counterexample causal graph, for which I can apply rule 3 on the intervention $\hat{x}$ in the query $P(y|\hat{x}, w)$, but I cannot use rule 1 on the observation $x$ in the query $P(y|x, w)$. Do the same for a graph where rule 2 is applicable on $\hat{x}$ but rule 1 on $x$ is not. Why is that so? Hint: think about the mutilations of graphs appearing in the rules of do-calculus]**

## Solution to Exercise 1

In the graph: the interventional query $P(y|\hat{x}, z_2)$ has 3 answers.




**Blocking variables:** $z_3, x$

$$
\begin{aligned}
P(y|\hat{x}, z_2) &= \sum_{z_3} P(y|\hat{x}, z_2, z_3) P(z_3|\hat{x}, z_2) \\
&= \sum_{z_3} P(y|\hat{x}, \hat{z_2}, z_3) P(z_3|\hat{x}, z_2) \\
&= \sum_{z_3} P(y|\hat{z_2}, z_3) P(z_3|\hat{x}, z_2) \\
&= \sum_{z_3, x'} \Big( P(y|\hat{z_2}, z_3, x') P(x'|\hat{z_2}, z_3) \Big) P(z_3|\hat{x}, z_2) \\
&= \sum_{z_3} P(z_3|x, z_2) \sum_{x'} P(y|z_2, z_3, x') P(x'|z_3) \\
&= \sum_{z_3} P(z_3|x, z_2) \sum_{x'} P(y|z_2, z_3, x') P(x')
\end{aligned} \tag{32}
$$

**Blocking variables:** $z_3, z_1$

$$
\begin{aligned}
P(y|\hat{x}, z_2) &= \sum_{z_3} P(y|\hat{x}, z_2, z_3)P(z_3|\hat{x}, z_2) \\
&= \sum_{z_3} P(y|\hat{x}, \hat{z}_2, z_3)P(z_3|\hat{x}, z_2) \\
&= \sum_{z_3} P(y|\hat{z}_2, z_3)P(z_3|\hat{x}, z_2) \\
&= \sum_{z_3, z_1} \Big( P(y|\hat{z}_2, z_3, z_1)P(z_1|\hat{z}_2, z_3) \Big) P(z_3|\hat{x}, z_2) \\
&= \sum_{z_3} P(z_3|x, z_2) \sum_{z_1} P(y|z_2, z_3, z_1)P(z_1|z_3) \\
&= \sum_{z_3} P(z_3|x, z_2) \sum_{z_1} P(y|z_2, z_3, z_1)P(z_1) \qquad (33)
\end{aligned}
$$

Note that the above results are basically identical, where instead of observing $x$, we observe its descendant $z_1$ (the paths in the graph are such that this is the only distinction between the two results, but this is not necessarily a general feature).
**Blocking variables:** $z_1, x$

$$
\begin{aligned}
P(y|\hat{x}, z_2) &= \sum_{z_1} P(y|\hat{x}, z_2, z_1)P(z_1|\hat{x}, z_2) \\
&= \sum_{z_1} P(y|\hat{x}, \hat{z}_1, z_2)P(z_1|\hat{x}, z_2) \\
&= \sum_{z_1} P(y|\hat{z}_1, z_2)P(z_1|\hat{x}, z_2) \\
&= \sum_{z_1, x'} \Big( P(y|\hat{z}_1, z_2, x')P(x'|\hat{z}_1, z_2) \Big) P(z_1|\hat{x}, z_2) \\
&= \sum_{z_1} P(z_1|x, z_2) \sum_{x'} P(y|z_1, z_2, x')P(x'|\hat{z}_1, \hat{z}_2) \\
&= \sum_{z_1} P(z_1|x, z_2) \sum_{x'} P(y|z_1, z_2, x')P(x'|\hat{z}_2) \\
&= \sum_{z_1} P(z_1|x, z_2) \sum_{x'} P(y|z_1, z_2, x')P(x') \qquad (34)
\end{aligned}
$$

## Solution to exercise 2

In the following simple graph:

$$
\begin{tikzcd}
 & z_3 \arrow[dl] \arrow[dr] & \\
x \arrow[r] & z_1 \arrow[d] & y \arrow[l] \\
 & z_2 &
\end{tikzcd}
$$

Let us consider the query: $P(y|\hat{x}, z_1)$ (the same arguments can be applied to the query $P(y|\hat{x}, z_2)$).

Now:

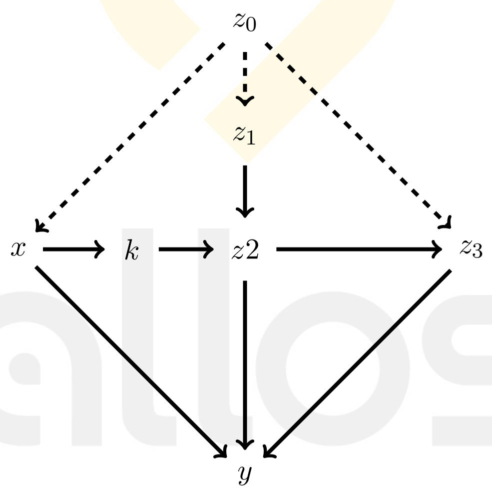
$$P(y|\hat{x}, z_1) = \sum_{z_3} P(y|\hat{x}, z_3, z_1) P(z_3|\hat{x}, z_1)$$
$$= \sum_{z_3} P(y|\hat{x}, z_3, z_1) \sum_{y'} P(z_3|\hat{x}, z_1, y') P(y'|\hat{x}, z_1) \quad (35)$$

Note now that 1) we are conditioning on $y$, but $y$ is the original outcome of our query (fishy); the last probability distribution on the right is *identical* to the original query on the left-hand side of the equation (very fishy).

**[Exercise 6: show that the above equality is not actually an infinite recursion, but rather a trivial identity $P(y|\hat{x}, z_1) = P(y|\hat{x}, z_1)$!!!]**

**[Exercise 7: now that we have seen our usual methods do not work, how can we re-write the query $P(y|\hat{x}, z_1)$ so that it is identifiable? what is the result? Hint: remember the decomposition of the joint probability of A and B, given the condition C, into a product of two (conditional) probabilities. Invert the identity to re-write the query above. Now solve it using the steps explained above for multiple outcomes queries.]**

**[Exercise 8: Try to answer the query $P(y|\hat{x})$ for the graph below (slightly modified version of Figure 1)**
**Prove that there are 3 solutions to this query, all requiring to condition on $z_2$ and $z_3$**




**first, then $(z_1, k)$, $(z_1, x)$ and finally $(k, x)$. Note in the last, case a subgraph similar to the one in Exercise 2 is analyzed, and as before, we cannot seem to be able to identify with usual methods one of the probabilities in the product. Use what you learned in exercise 6, then identify the above query when the conditioning variables are $(z_2, z_3, k, x)$.]**