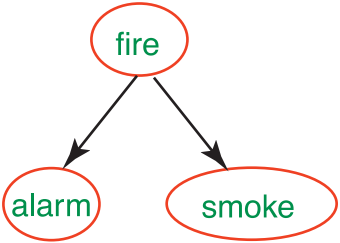
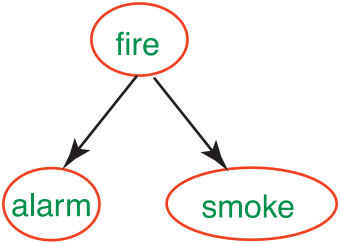
Understanding Independence: Common ancestors



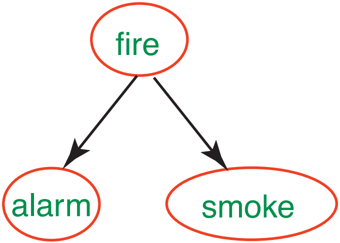
- *alarm* and *smoke* are

Understanding Independence: Common ancestors



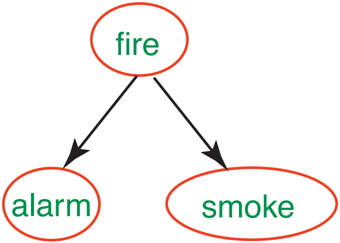
- *alarm* and *smoke* are dependent

Understanding Independence: Common ancestors



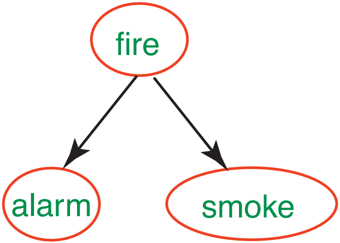
- *alarm* and *smoke* are dependent
- *alarm* and *smoke* are
  given *fire*

Understanding Independence: Common ancestors



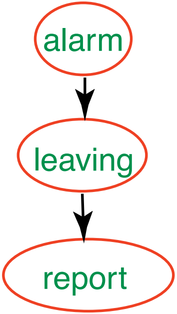
- *alarm* and *smoke* are dependent
- *alarm* and *smoke* are independent given *fire*

Understanding Independence: Common ancestors



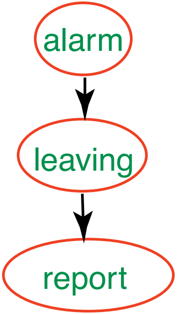
- *alarm* and *smoke* are dependent
- *alarm* and *smoke* are independent given *fire*
- Intuitively, *fire* can explain *alarm* and *smoke*; learning one can affect the other by changing your belief in *fire*.

Understanding Independence: Chain



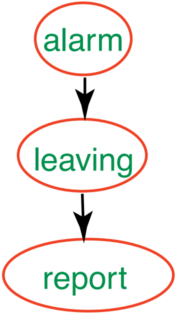
- *alarm* and *report* are

Understanding Independence: Chain



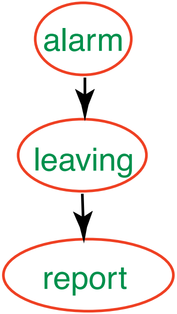
- *alarm* and *report* are dependent

Understanding Independence: Chain



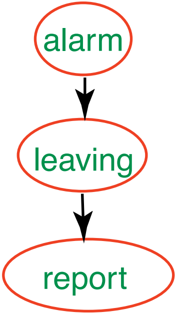
- *alarm* and *report* are dependent
- *alarm* and *report* are given *leaving*

Understanding Independence: Chain



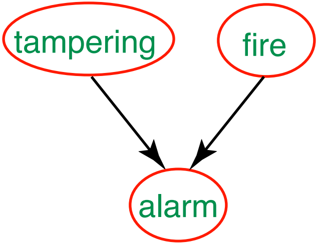
- *alarm* and *report* are dependent
- *alarm* and *report* are independent given *leaving*

Understanding Independence: Chain



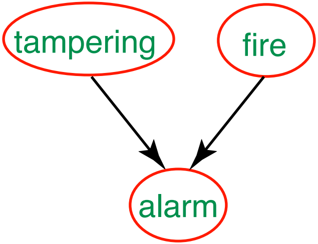
- *alarm* and *report* are dependent
- *alarm* and *report* are independent given *leaving*
- Intuitively, the only way that the *alarm* affects *report* is by affecting *leaving*.

Understanding Independence: Common descendants



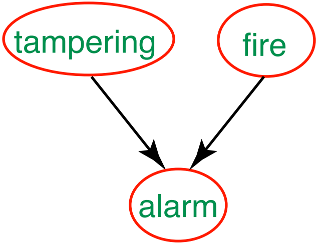
- *tampering* and *fire* are

Understanding Independence: Common descendants



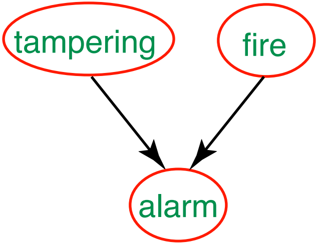
- *tampering* and *fire* are independent

Understanding Independence: Common descendants



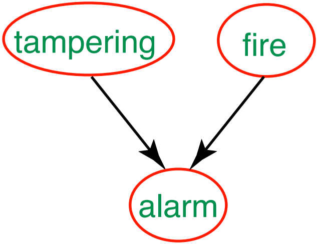
- *tampering* and *fire* are independent
- *tampering* and *fire* are given *alarm*

Understanding Independence: Common descendants



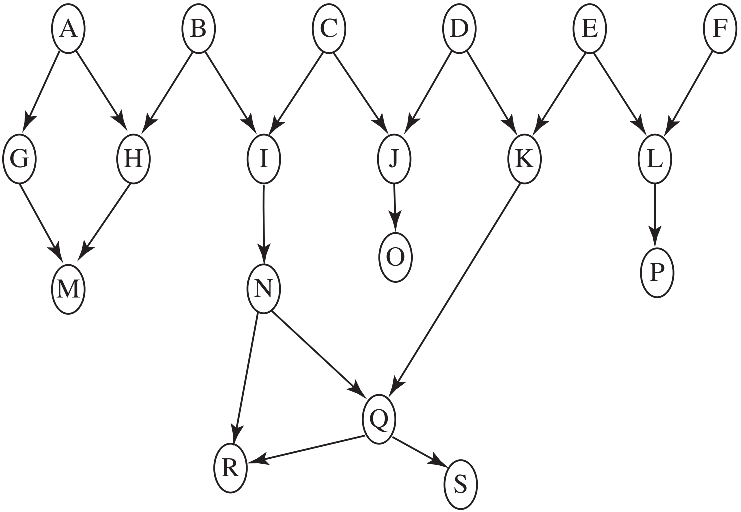
- *tampering* and *fire* are independent
- *tampering* and *fire* are dependent given *alarm*

Understanding Independence: Common descendants



- *tampering* and *fire* are independent
- *tampering* and *fire* are dependent given *alarm*
- Intuitively, *tampering* can explain away *fire*

Understanding independence: example



Understanding independence: questions

1. On which given probabilities does $P(N)$ depend?

Understanding independence: questions

1. On which given probabilities does $P(N)$ depend?
2. If you were to observe a value for B, which variables' probabilities will change?

Understanding independence: questions

1. On which given probabilities does $P(N)$ depend?
2. If you were to observe a value for B, which variables' probabilities will change?
3. If you were to observe a value for N, which variables' probabilities will change?

Understanding independence: questions

1. On which given probabilities does $P(N)$ depend?
2. If you were to observe a value for B, which variables' probabilities will change?
3. If you were to observe a value for N, which variables' probabilities will change?
4. Suppose you had observed a value for M; if you were to then observe a value for N, which variables' probabilities will change?

Understanding independence: questions

1. On which given probabilities does $P(N)$ depend?
2. If you were to observe a value for B, which variables' probabilities will change?
3. If you were to observe a value for N, which variables' probabilities will change?
4. Suppose you had observed a value for M; if you were to then observe a value for N, which variables' probabilities will change?
5. Suppose you had observed B and Q; which variables' probabilities will change when you observe N?

What variables are affected by observing?

- If you observe variable(s) $\overline{Y}$, the variables whose posterior probability is different from their prior are:
  - The ancestors of $\overline{Y}$ and
  - their descendants.
- Intuitively (if you have a causal belief network):
  - You do abduction to possible causes and
  - prediction from the causes.

d-separation

- A connection is a meeting of arcs in a belief network. A connection is open is defined as follows:
  - If there are arcs $A \rightarrow B$ and $B \rightarrow C$ such that $B \notin \overline{Z}$, then the connection at B between A and C is open.
  - If there are arcs $B \rightarrow A$ and $B \rightarrow C$ such that $B \notin \overline{Z}$, then the connection at B between A and C is open.
  - If there are arcs $A \rightarrow B$ and $C \rightarrow B$ such that B (or a descendent of B) is in $\overline{Z}$, then the connection at B between A and C is open.

d-separation

- A connection is a meeting of arcs in a belief network. A connection is open is defined as follows:
  - If there are arcs $A \rightarrow B$ and $B \rightarrow C$ such that $B \notin \overline{Z}$, then the connection at B between A and C is open.
  - If there are arcs $B \rightarrow A$ and $B \rightarrow C$ such that $B \notin \overline{Z}$, then the connection at B between A and C is open.
  - If there are arcs $A \rightarrow B$ and $C \rightarrow B$ such that B (or a descendent of B) is in $\overline{Z}$, then the connection at B between A and C is open.
- X is d-connected from Y given $\overline{Z}$ if there is a path from X to Y, along open connections.
- X is d-separated from Y given $\overline{Z}$ if it is not d-connected.

d-separation

- A connection is a meeting of arcs in a belief network. A connection is open is defined as follows:
  - If there are arcs $A \rightarrow B$ and $B \rightarrow C$ such that $B \notin \overline{Z}$, then the connection at B between A and C is open.
  - If there are arcs $B \rightarrow A$ and $B \rightarrow C$ such that $B \notin \overline{Z}$, then the connection at B between A and C is open.
  - If there are arcs $A \rightarrow B$ and $C \rightarrow B$ such that B (or a descendent of B) is in $\overline{Z}$, then the connection at B between A and C is open.
- X is d-connected from Y given $\overline{Z}$ if there is a path from X to Y, along open connections.
- X is d-separated from Y given $\overline{Z}$ if it is not d-connected.
- $\overline{X}$ is independent $\overline{Y}$ given $\overline{Z}$ for all conditional probabilities iff $\overline{X}$ is d-separated from $\overline{Y}$ given $\overline{Z}$

Markov Random Field

A Markov random field is composed of

- of a set of random variables: $X = \{X_1, X_2, \ldots, X_n\}$ and
- a set of factors $\{f_1, \ldots, f_m\}$, where a factor is a non-negative function of a subset of the variables.

Markov Random Field

A Markov random field is composed of

- of a set of random variables: $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$ and
- a set of factors $\{f_1, \ldots, f_m\}$, where a factor is a non-negative function of a subset of the variables.

and defines a joint probability distribution:

$$P(\mathbf{X} = \mathbf{x}) \propto \prod_k f_k(\mathbf{X}_k = \mathbf{x}_k) .$$

Markov Random Field

A Markov random field is composed of

- of a set of random variables: $\mathrm{X} = \{X_1, X_2, \ldots, X_n\}$ and
- a set of factors $\{f_1, \ldots, f_m\}$, where a factor is a non-negative function of a subset of the variables.

and defines a joint probability distribution:

$$
\begin{aligned}
P(\mathrm{X}=\mathrm{x}) &\propto \prod_k f_k(\mathrm{X}_k = \mathrm{x}_k)\,. \\
P(\mathrm{X}=\mathrm{x}) &= \frac{1}{Z}\prod_k f_k(\mathrm{X}_k = \mathrm{x}_k)\,. \\
Z &= \sum_{\mathrm{x}}\prod_k f_k(\mathrm{X}_k = \mathrm{x}_k)
\end{aligned}
$$

where $f_k(\mathrm{X}_k)$ is a factor on $\mathrm{X}_k \subseteq \mathrm{X}$, and
x_k is x projected onto X_k.
Z is a normalization constant known as the partition function.

Markov Networks and Factor graphs

- A Markov network is a graphical representation of a Markov random field where the nodes are the random variables and there is an arc between any two variables that are in a factor together.

Markov Networks and Factor graphs

- A Markov network is a graphical representation of a Markov random field where the nodes are the random variables and there is an arc between any two variables that are in a factor together.
- A factor graph is a bipartite graph, which contains a variable node for each random variable and a factor node for each factor. There is an edge between a variable node and a factor node if the variable appears in the factor.

Markov Networks and Factor graphs

- A Markov network is a graphical representation of a Markov random field where the nodes are the random variables and there is an arc between any two variables that are in a factor together.
- A factor graph is a bipartite graph, which contains a variable node for each random variable and a factor node for each factor. There is an edge between a variable node and a factor node if the variable appears in the factor.
- A belief network is a

Markov Networks and Factor graphs

- A Markov network is a graphical representation of a Markov random field where the nodes are the random variables and there is an arc between any two variables that are in a factor together.
- A factor graph is a bipartite graph, which contains a variable node for each random variable and a factor node for each factor. There is an edge between a variable node and a factor node if the variable appears in the factor.
- A belief network is a type of Markov random field where the factors represent conditional probabilities, there is a factor for each variable, and directed graph is acyclic.

Independence in a Markov Network

- The Markov blanket of a variable X is the set of variables that are in factors with X.
- A variable is independent of the other variables given its Markov blanket.

Independence in a Markov Network

- The Markov blanket of a variable X is the set of variables that are in factors with X.
- A variable is independent of the other variables given its Markov blanket.
- X is connected to Y given $\overline{Z}$ if there is a path from X to Y in the Markov network, which does not contain an element of Z.
- X is separated from Y given $\overline{Z}$ if it is not connected.

Independence in a Markov Network

- The Markov blanket of a variable X is the set of variables that are in factors with X.
- A variable is independent of the other variables given its Markov blanket.
- X is connected to Y given $\overline{Z}$ if there is a path from X to Y in the Markov network, which does not contain an element of Z.
- X is separated from Y given $\overline{Z}$ if it is not connected.
- A positive distribution is one that does not contain zero probabilities.
- $\overline{X}$ is independent $\overline{Y}$ given $\overline{Z}$ for all positive distributions iff $\overline{X}$ is separated from $\overline{Y}$ given $\overline{Z}$

Canonical Representations

- The parameters of a graphical model are the numbers that define the model.
- A belief network is a canonical representation: given the structure and the distribution, the parameters are uniquely determined.
- A Markov random field is not a canonical representation. Many different parameterizations result in the same distribution.

Representations of Conditional Probabilities

There are many representations of conditional probabilities and factors:

Representations of Conditional Probabilities

There are many representations of conditional probabilities and factors:

- Tables
- Decision Trees
- Rules
- Weighted Logical Formulae
- Noisy-or
- Logistic Function
- Neural network

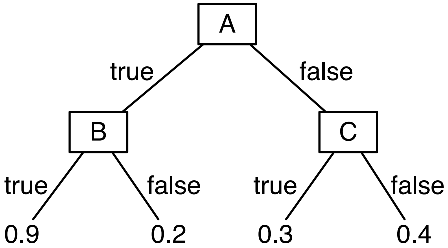
Tabular Representation

$P(D \mid A, B, C)$:

A	B	C	D	Prob
true	true	true	true	0.9
true	true	true	false	0.1
true	true	false	true	0.9
true	true	false	false	0.1
true	false	true	true	0.2
true	false	true	false	0.8
true	false	false	true	0.2
true	false	false	false	0.8
false	true	true	true	0.3
false	true	true	false	0.7
false	true	false	true	0.4
false	true	false	false	0.6
false	false	true	true	0.3
false	false	true	false	0.7
false	false	false	true	0.4
false	false	false	false	0.6

Decision Tree Representation

$$P(d \mid A, B, C)$$



Rule Representation

$$0.9 : \; d \leftarrow a \wedge b$$

$$0.2 : \; d \leftarrow a \wedge \neg b$$

$$0.3 : \; d \leftarrow \neg a \wedge c$$

$$0.4 : \; d \leftarrow \neg a \wedge \neg c$$

Weighted Logical Formulae

$$\begin{aligned} d \leftrightarrow ((a \wedge b \wedge n_0) \\ \vee (a \wedge \neg b \wedge n_1) \\ \vee (\neg a \wedge c \wedge n_2) \\ \vee (\neg a \wedge \neg c \wedge n_3)) \end{aligned}$$

n_i are independent:

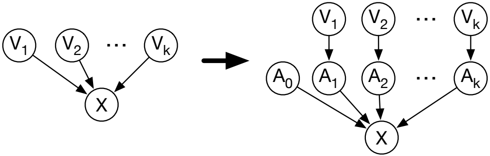
$$\begin{aligned} P(n_0) &= 0.9 \\ P(n_1) &= 0.2 \\ P(n_2) &= 0.3 \\ P(n_3) &= 0.4 \end{aligned}$$

Noisy-or

The robot is wet if it gets wet from rain or coffee or sprinkler or another reason. They each have a probability of making the robot wet $\longrightarrow$ noisy-or.

Noisy-or

The robot is wet if it gets wet from rain or coffee or sprinkler or another reason. They each have a probability of making the robot wet $\longrightarrow$ noisy-or.



X has Boolean parents $V_1 \dots V_k$, $\longrightarrow$ $k+1$ parameters $p_0 \dots p_k$.
invent Boolean variables $A_0, A_1, \dots, A_k$,
with probabilities $P(A_0) = p_0$ and for $i > 0$

$$P(A_i{=}true \mid V_i{=}true) = p_i$$
$$P(A_i{=}true \mid V_i{=}false) = 0$$

$$P(X \mid A_0, A_1, \dots, A_k) = \begin{cases} 1 & \text{if } \exists i \; A_i \text{ is true} \\ 0 & \text{if } \forall i \; A_i \text{ is false} \end{cases}$$

Noisy-or: example

- Suppose the robot could get wet from rain or coffee.
- There is a probability that it gets wet from rain if it rains, and a probability that it gets wet from coffee if it has coffee, and a probability that it gets wet for other reasons.

Noisy-or: example

- Suppose the robot could get wet from rain or coffee.
- There is a probability that it gets wet from rain if it rains, and a probability that it gets wet from coffee if it has coffee, and a probability that it gets wet for other reasons.
- We could have:
  $P(wet\_from\_rain \mid rain) = 0.3$,
  $P(wet\_from\_coffee \mid coffee) = 0.2$
  $P(wet\_for\_other\_reasons) = 0.1$.

Noisy-or: example

- Suppose the robot could get wet from rain or coffee.
- There is a probability that it gets wet from rain if it rains, and a probability that it gets wet from coffee if it has coffee, and a probability that it gets wet for other reasons.
- We could have:
  $P(wet\_from\_rain \mid rain) = 0.3$,
  $P(wet\_from\_coffee \mid coffee) = 0.2$
  $P(wet\_for\_other\_reasons) = 0.1$.
- The robot is wet if it wet from rain, wet from coffee, or wet for other reasons.

$$wet \leftrightarrow wet\_from\_rain \vee wet\_from\_coffe \vee wet\_for\_other\_reasons$$

Logistic Functions

$$P(h \mid e) = \frac{P(h \wedge e)}{P(e)}$$

Logistic Functions

$$
\begin{aligned}
P(h \mid e) &= \frac{P(h \wedge e)}{P(e)} \\
&= \frac{P(h \wedge e)}{P(h \wedge e) + P(\neg h \wedge e)}
\end{aligned}
$$

Logistic Functions

$$
\begin{aligned}
P(h \mid e) &= \frac{P(h \wedge e)}{P(e)} \\
&= \frac{P(h \wedge e)}{P(h \wedge e) + P(\neg h \wedge e)} \\
&= \frac{1}{1 + P(\neg h \wedge e)/P(h \wedge e)}
\end{aligned}
$$

Logistic Functions

$$
\begin{aligned}
P(h \mid e) &= \frac{P(h \wedge e)}{P(e)} \\
&= \frac{P(h \wedge e)}{P(h \wedge e) + P(\neg h \wedge e)} \\
&= \frac{1}{1 + P(\neg h \wedge e) / P(h \wedge e)} \\
&= \frac{1}{1 + e^{-\log P(h \wedge e)/P(\neg h \wedge e)}}
\end{aligned}
$$

Logistic Functions

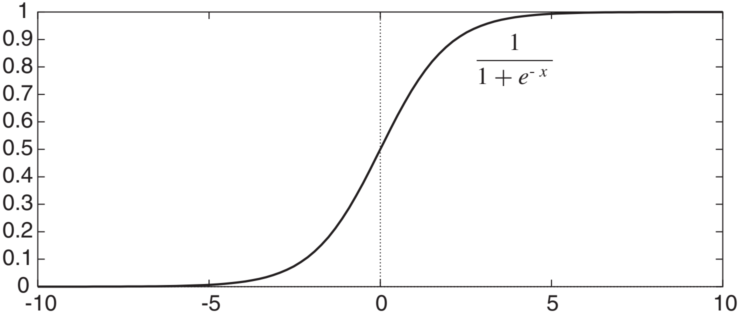
$$
\begin{aligned}
P(h \mid e) &= \frac{P(h \wedge e)}{P(e)} \\
&= \frac{P(h \wedge e)}{P(h \wedge e) + P(\neg h \wedge e)} \\
&= \frac{1}{1 + P(\neg h \wedge e)/P(h \wedge e)} \\
&= \frac{1}{1 + e^{-\log P(h \wedge e)/P(\neg h \wedge e)}} \\
&= sigmoid(\log odds(h \mid e))
\end{aligned}
$$

$$sigmoid(x) = \frac{1}{1 + e^{-x}}$$

$$odds(h \mid e) = \frac{P(h \wedge e)}{P(\neg h \wedge e)}$$

Logistic Functions

A conditional probability is the sigmoid of the log-odds.



$$sigmoid(x) = \frac{1}{1+e^{-x}}$$

A logistic function is the sigmoid of a linear function.

Logistic Representation of Conditional Probability

$$\begin{aligned}P(d \mid A, B, C) = \mathit{sigmoid}(&0.9^{\dagger} * A * B \\ &+ 0.2^{\dagger} * A * (1 - B) \\ &+ 0.3^{\dagger} * (1 - A) * C \\ &+ 0.4^{\dagger} * (1 - A) * (1 - C))\end{aligned}$$

where $0.9^{\dagger}$ is $\mathit{sigmoid}^{-1}(0.9)$.

Logistic Representation of Conditional Probability

$$\begin{aligned} P(d \mid A, B, C) = \mathit{sigmoid}(&0.9^\dagger * A * B \\ &+ 0.2^\dagger * A * (1 - B) \\ &+ 0.3^\dagger * (1 - A) * C \\ &+ 0.4^\dagger * (1 - A) * (1 - C)) \end{aligned}$$

where $0.9^\dagger$ is $\mathit{sigmoid}^{-1}(0.9)$.

$$\begin{aligned} P(d \mid A, B, C) = \mathit{sigmoid}(&0.4^\dagger \\ &+ (0.2^\dagger - 0.4^\dagger) * A \\ &+ (0.9^\dagger - 0.2^\dagger) * A * B \\ &+ \dots \end{aligned}$$

Neural Network

- Build a neural network to predict D from A, B, C

Neural Network

- Build a neural network to predict D from A, B, C
- A neural network tries to predict expected values

Neural Network

- Build a neural network to predict D from A, B, C
- A neural network tries to predict expected values
- The expected value of a variable with domain $\{0, 1\}$ is its probability.

Neural Network

- Build a neural network to predict D from A, B, C
- A neural network tries to predict expected values
- The expected value of a variable with domain $\{0, 1\}$ is its probability.
  Typically use a sigmoid activation at the root, and optimize with log likelihood.

Neural Network

- Build a neural network to predict D from A, B, C
- A neural network tries to predict expected values
- The expected value of a variable with domain $\{0, 1\}$ is its probability.
  Typically use a sigmoid activation at the root, and optimize with log likelihood.
  — such a neural network is a logistic regression model with learned features
- For other discrete variables, the expected value is *not* the probability.

Neural Network

- Build a neural network to predict D from A, B, C
- A neural network tries to predict expected values
- The expected value of a variable with domain $\{0, 1\}$ is its probability.
  Typically use a sigmoid activation at the root, and optimize with log likelihood.
  — such a neural network is a logistic regression model with learned features
- For other discrete variables, the expected value is *not* the probability.
  We create a Boolean ($\{0, 1\}$) variable for each value — indicator variable $\equiv$ having an output for each value

Neural Network

- Build a neural network to predict D from A, B, C
- A neural network tries to predict expected values
- The expected value of a variable with domain $\{0, 1\}$ is its probability.
  Typically use a sigmoid activation at the root, and optimize with log likelihood.
  — such a neural network is a logistic regression model with learned features
- For other discrete variables, the expected value is *not* the probability.
  We create a Boolean ($\{0, 1\}$) variable for each value — indicator variable $\equiv$ having an output for each value
- For other domains, a Bayesian neural network can represent the distribution over the outputs (not just a point prediction).