6. (20 points.) **Classification**

Consider a traffic-monitoring agent trying to decide whether the stoplights at an intersection are working or not ($W = w$ or $\neg w$. The agent observes two variables, the East-West light's color EW, which is either *red* or *green*, and the North-South light's color NS, which is also either red or green. When the lights are functioning, exactly one of the lights is green, while when the lights are broken, both are red (and flashing, but the agent cannot perceive this distinction). The agent has several observations as training data:

NS	EW	W
r	g	w
g	r	w
r	r	$\neg w$
g	r	w
r	g	w
r	g	w
g	r	w

(a) **(4 pts)** Assume the agent uses a naive Bayes model to make its predictions. Based on the training data above, fill in the CPTs below (use the maximum likelihood estimates, i.e. no smoothing).

prior $P(W=w) = \frac{6}{7}$

$P(NS=r \mid W=w)$

		$W=w$	$W=\neg w$
NS	r	$1/2$	1
	g	$1/2$	0

		$W=w$	$W=\neg w$
EW	r	$1/2$	1
	g	$1/2$	0

(b) **(6 pts)** Assume the agent observes that both lights are red. What is the posterior probability that the lights are working *according to the agent's model.*

$$P(W \mid NS=r, EW=r) \propto \begin{pmatrix} \frac{1}{2}\cdot\frac{1}{2}\cdot\frac{6}{7} \\ 1\cdot 1\cdot\frac{1}{7} \end{pmatrix} = \begin{pmatrix} \frac{3}{14} \\ \frac{2}{14} \end{pmatrix}$$

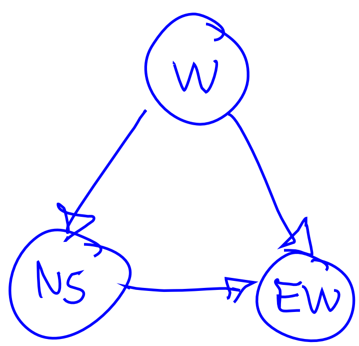
renormalize to get

$$P(W=w \mid NS=r, EW=r) = 3/5$$

(c) **(3 pts)** Note that in the data, all red/red data points are from broken lights. Explain what critical conditional dependence the model fails to capture.

Both lights are correlated (i.e. dependent of each other).

(d) **(3 pts)** Draw a Bayes' net which would allow the agent to draw the correct inference in the case of two red lights (as well as the working light cases). You should not need to introduce any new variables.



(e) **(4 pts)** Give a minimal set of features which would be sufficient for a perceptron to correctly predict W on all training examples. State your features precisely as functions from inputs $x = (ew, ns)$ to real numbers.

$$\text{different}(NS, EW) = \begin{cases} +1 & \text{if } NS \neq EW \\ -1 & \text{o.w.} \end{cases}$$

then $\left.\begin{matrix} w_{w} = 1 \\ w_{\neg w} = -1 \end{matrix}\right\}$ separates data

5 *Classification*

Imagine we have features f_1, f_2, f_3, f_4 and three classes, $\{x, y, z\}$. Assume we are training a multi-class perceptron and a given point in the training, it has the following weight vectors:

$$
\begin{aligned}
w_x &= \langle 0, 0, 0, 0 \rangle \\
w_y &= \langle 0, 2, 0, 0 \rangle \\
w_z &= \langle 2, 0, 1, 0 \rangle
\end{aligned}
$$

(a) If we next encounter the instance $\langle 1, 0, 0, 1 \rangle$ with true label x, write the resulting weights after processing this new instance:

Score $(x) = 0$

" $(y) = 0$ $\Rightarrow$ $y_{pred} = z$

" $(z) = 2$ $y_{truth} = x$

So $W_z^{new} = W_z^{old} - \langle 1,0,0,1 \rangle = \langle 1, 0, 1, -1 \rangle$

$W_x^{new} = W_x^{old} + \langle 1,0,0,1 \rangle = \langle 1,0,0,1 \rangle$

$W_y^{new} = W_y^{old} = \langle 0, 2, 0, 0 \rangle$

(b) If we next encounter the instance $\langle 0, 1, 1, 1 \rangle$ with true label y, write the resulting weights after processing this new instance: (from original weights - forget about update in a)

Score $(x) = 0$

Score $(y) = 2$ $\Rightarrow$ $y_{pred} = y_{truth} = y$

Score $(z) = 1$

So do nothing! (i.e. $W_{y'}^{new} = W_{y'}^{old}$ $\forall$ labels y')