计算机科学188：人工智能导论
2024年春季 笔记16

作者（其他所有笔记）：尼基尔·夏尔马
作者（贝叶斯网络笔记）：乔希·胡格和杰基·梁，由王蕾吉娜编辑
作者（逻辑笔记）：亨利·朱，由考佩林编辑
致谢（机器学习与逻辑笔记）：部分章节改编自教材《人工智能：一种现代方法》。

最后更新时间：2023年8月26日

粒子滤波

回想一下，对于贝叶斯网络，当进行精确推理的计算成本过高时，使用我们讨论过的采样技术之一是有效近似所需概率分布的可行替代方法。隐马尔可夫模型也有同样的缺点——使用前向算法进行精确推理所需的时间与随机变量域中的值数量成正比。在我们当前的天气问题表述中，这是可以接受的，因为天气只能取2个值，$W_i \in \{\text{晴天}, \text{雨天}\}$，但假设我们想进行推理以计算给定日期实际温度分布到最接近的十分之一度。

与贝叶斯网络采样类似的隐马尔可夫模型被称为粒子滤波，它涉及通过状态图模拟一组粒子的运动，以近似所讨论随机变量的概率（信念）分布。这与前向算法解决的是同一个问题：它为我们提供了 $P\left(X_N \mid e_{1:N}\right)$ 的近似值。

我们不会存储将每个状态映射到其信念概率的完整概率表，而是存储 n 个粒子的列表，其中每个粒子处于我们随时间变化的随机变量域中的 d 个可能状态之一。通常，n 远小于 d（用符号表示为 $n << d$），但仍足够大以产生有意义的近似值；否则，粒子滤波的性能优势就会变得微不足道。在该算法中，粒子只是样本的名称。

我们认为，在任何给定的时间步长，一个粒子处于任何给定状态，这完全取决于在我们的模拟中该时间步长处于该状态的粒子数量。例如，假设我们确实想要模拟某一天 T 的温度 i 的信念分布，为简单起见，假设这个温度只能取 $[10, 20]$ 范围内的整数值（$d = 11$ 种可能状态）。进一步假设我们有 $n = 10$ 个粒子，在我们模拟的时间步长 i 时取以下值：

$$[15, 12, 12, 10, 18, 14, 12, 11, 11, 10]$$

通过统计出现在我们粒子列表中的每个温度，并除以粒子总数，我们可以生成我们想要的时间 i 时温度的经验分布：

T_i	10	11	12	13	14	15	16	17	18	19	20
$B(T_i)$	0.2	0.2	0.3	0	0.1	0.1	0	0	0.1	0	0

既然我们已经了解了如何从粒子列表中恢复信念分布，那么接下来要讨论的就是如何为我们选择的时间步生成这样一个列表。

粒子滤波模拟

粒子滤波模拟从粒子初始化开始，这可以非常灵活地完成——我们可以随机、均匀地采样粒子，或者从某个初始分布中采样。一旦我们采样了一个初始粒子列表，模拟就会采用与前向算法类似的形式，在每个时间步进行时间推移更新，然后进行观测更新：

•时间推移更新——根据转移模型更新每个粒子的值。对于处于状态 t_i 的粒子，从由 $P\left(T_{i+1} \mid t_i\right)$ 给出的概率分布中采样更新后的值。注意时间推移更新与使用贝叶斯网络进行先验采样的相似性，因为任何给定状态下粒子的频率反映了转移概率。

•观测更新 - 在粒子滤波的观测更新过程中，我们使用传感器模型 $P\left(F_i \mid T_i\right)$ 根据观测证据和粒子状态所决定的概率对每个粒子进行加权。具体而言，对于处于状态 t_i 且传感器读数为 f_i 的粒子，赋予权重 $P\left(f_i \mid t_i\right)$ 。观测更新的算法如下：

1. 按照上述方法计算所有粒子的权重。
2. 计算每个状态的总权重。
3. 如果所有状态的权重总和为0，则重新初始化所有粒子。
4. 否则，对状态上的总权重分布进行归一化，并从该分布中对粒子列表进行重采样。

注意观测更新与似然加权的相似性，在似然加权中，我们同样根据证据对样本进行降权。

让我们通过示例来看看是否能更好地理解这个过程。为我们的天气场景定义一个转移模型，将温度用作随时间变化的随机变量，如下所示：对于特定的温度状态，你可以保持在同一状态，或者在 $[10, 20]$ 范围内转移到相差一度的状态。在所有可能的结果状态中，转移到最接近15的状态的概率为 80% ，其余的结果状态将剩余的 20% 概率平均分配。

我们的温度粒子列表如下：

$$[15, 12, 12, 10, 18, 14, 12, 11, 11, 10]$$

要对该粒子列表中处于 $T_i = 15$ 状态的第一个粒子进行时间推移更新，我们需要相应的转移模型：

T_{i+1}	14	15	16
$P(T_{i+1}\|T_i = 15)$	0.1	0.8	0.1

在实际操作中，我们为 T_{i+1} 域中的每个值分配不同的值范围，使得这些范围完全覆盖区间 $[0, 1)$ 且不重叠。对于上述转移模型，范围如下：

1. $T_{i+1} = 14$ 的范围是 $0 \leq r < 0.1$。
2. $T_{i+1} = 15$ 的范围是 $0.1 \leq r < 0.9$。
3. $T_{i+1} = 16$ 的范围是 $0.9 \leq r < 1$。

为了对处于状态 $T_i = 15$ 的粒子进行重采样，我们只需在 $[0, 1)$ 范围内生成一个随机数，然后查看它落在哪个区间。因此，如果我们的随机数是 $r = 0.467$，那么处于 $T_i = 15$ 的粒子就会留在 $T_{i+1} = 15$ 中，因为 $0.1 \leq r < 0.9$。现在考虑区间 $[0, 1)$ 内的以下10个随机数列表：

$$[0.467, 0.452, 0.583, 0.604, 0.748, 0.932, 0.609, 0.372, 0.402, 0.026]$$

如果我们使用这10个值作为对10个粒子进行重采样的随机值，那么在经过完整的时间步更新后，我们的新粒子列表应该如下所示：

$$[15, 13, 13, 11, 17, 15, 13, 12, 12, 10]$$

自己验证一下！更新后的粒子列表会产生相应的更新后的信念分布 $B(T_{i+1})$：

T_i	10	11	12	13	14	15	16	17	18	19	20
$B(T_{i+1})$	0.1	0.1	0.2	0.3	0	0.2	0	0.1	0	0	0

将我们更新后的信念分布 $B(T_{i+1})$ 与初始信念分布 $B(T_i)$ 进行比较，我们可以看到，总体趋势是粒子倾向于收敛到 $T = 15$ 的温度。

接下来，假设我们的传感器模型 $P(F_i \mid T_i)$ 表明正确预测 $f_i = t_i$ 的概率为 80%，且预测其他10种状态中任何一种的概率均等为 2%，让我们进行观测更新。假设预测为 $F_{i+1} = 13$，我们的10个粒子的权重如下：

Particle	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
State	15	13	13	11	17	15	13	12	12	10
Weight	0.02	0.8	0.8	0.02	0.02	0.02	0.8	0.02	0.02	0.02

然后我们按状态汇总权重：

State	10	11	12	13	15	17
Weight	0.02	0.02	0.04	2.4	0.04	0.02

所有权重值的总和为2.54，我们可以通过将每个条目除以这个总和来对权重表进行归一化，以生成概率分布：

State	10	11	12	13	15	17
Weight	0.02	0.02	0.04	2.4	0.04	0.02
Normalized Weight	0.0079	0.0079	0.0157	0.9449	0.0157	0.0079

最后一步是从这个概率分布中进行重采样，使用我们在时间推移更新期间用于重采样的相同技术。假设我们在 $[0, 1)$ 范围内生成10个随机数，其值如下：

$$[0.315, 0.829, 0.304, 0.368, 0.459, 0.891, 0.282, 0.980, 0.898, 0.341]$$

这会产生如下重采样粒子列表：

$$[13, 13, 13, 13, 13, 13, 13, 15, 13, 13]$$

以及相应的最终新信念分布：

T_i	10	11	12	13	14	15	16	17	18	19	20
$B(T_{i+1})$	0	0	0	0.9	0	0.1	0	0	0	0	0

注意，我们的传感器模型编码了我们的天气预报有80%的概率非常准确，并且我们的新粒子列表与此一致，因为大多数粒子被重采样为 $T_{i+1} = 13$ 。

效用

在我们对理性智能体的讨论中，效用的概念反复出现。例如，在游戏中，效用值通常被硬编码到游戏中，智能体使用这些效用值来选择行动。我们现在将讨论生成一个可行的效用函数需要什么。

理性智能体必须遵循最大效用原则——它们必须始终选择能使预期效用最大化的行动。然而，遵循这一原则仅对具有理性偏好的智能体有益。为了构建一个非理性偏好的例子，假设有3个物体， A, B 、C ，并且我们的智能体当前拥有 A 。假设我们的智能体有以下一组非理性偏好：

- 我们的智能体更喜欢 B 而不是 A 加 $\$1$
- 我们的智能体更喜欢 C 而不是 B 加 $\$1$
- 我们的智能体更喜欢 A 而不是 C 加 $\$1$

一个持有 B 和 C 的恶意智能体可以用 B 从我们的智能体那里换得 A 再加一美元，然后用 C 换得 B 再加一美元，接着再用 A 换得 C 再加一美元。我们的智能体就这样白白损失了 $\$3$ ！ 这样一来，我们的智能体可能会被迫在一个无尽的噩梦般的循环中放弃所有的钱。
现在让我们正式定义偏好的数学语言：

- 如果一个智能体更喜欢获得奖品 A 而不是获得奖品 B ，这写作 $A \succ B$
- 如果一个主体在接受 A 或 B 之间无差异，这记为 $A \sim B$
- 彩票是一种具有不同奖品且以不同概率出现的情形。为表示以概率 p 获得 A 且以概率(1 - p)获得 B 的彩票，我们记为

$$L = [p, A;\ (1-p), B]$$

为使一组偏好是理性的，它们必须遵循理性的五条公理：

- 可排序性： $(A \succ B) \vee (B \succ A) \vee (A \sim B)$
  一个理性主体必须要么偏好 A 或 B 中的一个，要么在两者之间无差异。
- 传递性： $(A \succ B) \wedge (B \succ C) \Rightarrow (A \succ C)$
  如果一个理性主体偏好 A 甚于 B 且偏好 B 甚于 C ，那么它偏好 A 甚于 C 。

- 连续性： $A \succ B \succ C \Rightarrow \exists p\,[p, A; (1-p), C] \sim B$
  如果一个理性主体偏好 A 甚于 B，但偏好 B 甚于 C，那么通过适当选择 p，有可能在 A 和 C 之间构造一个彩票 L，使得该主体对 L 和 B 无差异。

- 可替代性： $A \sim B \Rightarrow [p, A; (1-p), C] \sim [p, B; (1-p), C]$
  一个对两个奖品 A 和 B 无差异的理性主体，对于任何仅在将 A 替换为 B 或将 B 替换为 A 方面存在差异的两个彩票也无差异。

- 单调性： $A \succ B \Rightarrow (p \geq q \Leftrightarrow [p, A; (1-p), B] \succcurlyeq [q, A; (1-q), B]$
  如果一个理性主体偏好 A 甚于 B，那么在仅涉及 A 和 B 的彩票选择中，该主体会偏好将最高概率分配给 A 的彩票。

如果一个主体满足所有五个公理，那么可以保证该主体的行为可以描述为预期效用的最大化。更具体地说，这意味着存在一个实值效用函数 U，当应用该函数时，会为更偏好的奖品分配更高的效用，并且彩票的效用是彩票结果奖品效用的期望值。这两个陈述可以用两个简洁的数学等式总结如下：

$$U(A) \geq U(B) \quad \Leftrightarrow \quad A \succeq B \tag{1}$$

$$U([p_1, S_1;\ \ldots\ ; p_n, S_n]) \quad = \quad \sum_i p_i U(S_i) \tag{2}$$

如果满足这些约束条件并做出合适的算法选择，那么实现这种效用函数的智能体就保证会表现得最优。让我们通过一个具体例子更详细地讨论效用函数。考虑以下彩票：

$$L = [0.5, \$0;\ 0.5, \$1000]$$

这代表一种彩票，你有0.5的概率获得 $\$1000$，有0.5的概率获得 $\$0$。现在考虑三个智能体 A_1, A_2、 A_3，它们分别具有效用函数 $U_1(\$x) = x, U_2(\$x) = \sqrt{x}$、 $U_3(\$x) = x^2$。如果这三个智能体中的每一个都面临参与彩票和接受 $\$500$ 的固定支付之间的选择，它们会选择哪一个？参与彩票和接受固定支付对每个智能体的各自效用列于下表：

Agent	Lottery	Flat Payment
1	500	500
2	15.81	22.36
3	500000	250000

利用上述公式(2)，计算出了这些彩票的效用值：

$$U_1(L) \quad = \quad U_1([0.5, \$0;\ 0.5, \$1000]) = 0.5 \cdot U_1(\$1000) + 0.5 \cdot U_1(\$0) = 0.5 \cdot 1000 + 0.5 \cdot 0 = \boxed{500}$$

$$U_2(L) \quad = \quad U_2([0.5, \$0;\ 0.5, \$1000]) = 0.5 \cdot U_2(\$1000) + 0.5 \cdot U_2(\$0) = 0.5 \cdot \sqrt{1000} + 0.5 \cdot \sqrt{0} = \boxed{15.81}$$

$$U_3(L) \quad = \quad U_1([0.5, \$0;\ 0.5, \$1000]) = 0.5 \cdot U_3(\$1000) + 0.5 \cdot U_3(\$0) = 0.5 \cdot 1000^2 + 0.5 \cdot 0^2 = \boxed{500000}$$

根据这些结果，我们可以看出，主体 A_1 对参与彩票和接受固定报酬无差异（两种情况下的效用相同）。这样的主体被称为风险中性。类似地，主体 A_2 偏好固定报酬而非彩票，被称为风险厌恶型，而主体 A_3 偏好彩票而非固定报酬，被称为风险寻求型。

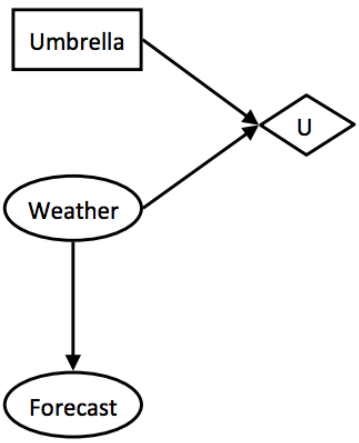
决策网络

在第三篇笔记中，我们学习了博弈树以及诸如极小极大算法和期望极大化算法等，我们使用这些算法来确定能使我们的期望效用最大化的最优行动。然后在第五篇笔记中，我们讨论了贝叶斯网络以及如何利用我们已知的证据进行概率推理以做出预测。现在我们将讨论一种结合了贝叶斯网络和期望极大化算法的决策网络，我们可以基于一个总体的图形概率模型，用它来对各种行动对效用的影响进行建模。让我们直接深入了解决策网络的结构：

- 机会节点——决策网络中的机会节点与贝叶斯网络中的行为相同。机会节点中的每个结果都有一个相关的概率，这可以通过对其所属的底层贝叶斯网络进行推理来确定。我们将用椭圆形表示这些节点。
- 行动节点——行动节点是我们可以完全控制的节点；它们代表我们有权从多个行动中进行选择的节点。我们将用矩形表示行动节点。

- 效用节点 - 效用节点是动作节点和机会节点某种组合的子节点。它们根据父节点所取的值输出一个效用，并在我们的决策网络中表示为菱形。

考虑这样一种情况：早上你准备去上课的时候，正在决定是否带伞，并且你知道天气预报有 30% 的降雨概率。你应该带伞吗？如果降雨概率是80%，你的答案会改变吗？这种情况非常适合用决策网络进行建模，我们按如下方式进行：



正如我们在本课程中对所讨论的各种建模技术和算法所做的那样，我们使用决策网络的目标再次是选择产生最大期望效用（MEU）的行动。这可以通过一个相当直接且直观的过程来完成：

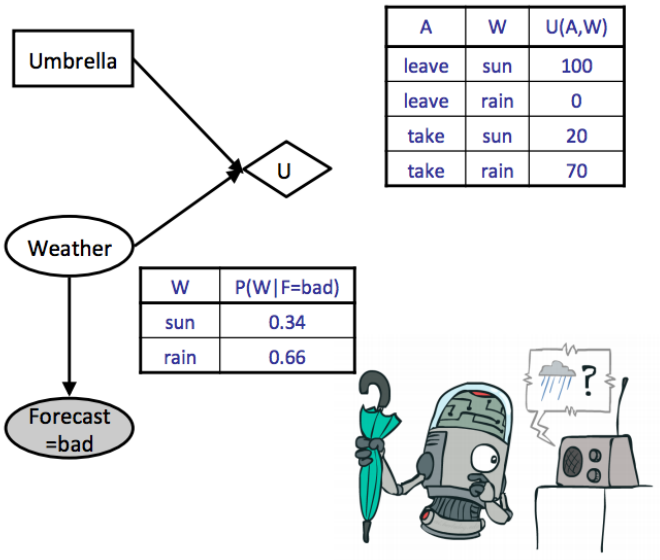
- 首先实例化所有已知证据，并运行推理以计算行动节点所指向的效用节点的所有机会节点父节点的后验概率。
- 遍历每个可能的行动，并根据上一步计算出的后验概率计算采取该行动的期望效用。给定证据 a 以及 e 和 n 机会节点，采取行动 a 的期望效用使用以下公式计算：

$$EU(a|e) = \sum_{x_1,\ldots,x_n} P(x_1,\ldots,x_n|e)U(a,x_1,\ldots,x_n)$$

其中每个 x_i 代表 i^{th} 机会节点可以取的值。我们只需在给定行动下对每个结果的效用进行加权求和，权重对应于每个结果的概率。

- 最后，选择产生最高效用的行动以得到MEU。

让我们通过计算天气示例中的最优行动（我们应该留下还是带上雨伞）来看看实际情况如何，这里会用到给定恶劣天气预报（天气预报是我们的证据变量）时天气的条件概率表，以及给定我们的行动和天气时的效用表：



A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

W	P(W|F=bad)
sun	0.34
rain	0.66

请注意，我们省略了后验概率 $P\left(W \mid F = \text{bad}\right)$ 的推理计算，但我们可以使用我们为贝叶斯网络讨论的任何推理算法来计算这些概率。相反，这里我们简单地假设上述给定的 $P\left(W \mid F = \text{bad}\right)$ 后验概率表。遍历我们的两个行动并计算期望效用，结果如下：

$$
\begin{aligned}
EU(\text{leave}|\text{bad}) &= \sum_w P(w|\text{bad})U(\text{leave},w) \\
&= 0.34 \cdot 100 + 0.66 \cdot 0 = \boxed{34}
\end{aligned}
$$

$$
\begin{aligned}
EU(\text{take}|\text{bad}) &= \sum_w P(w|\text{bad})U(\text{take},w) \\
&= 0.34 \cdot 20 + 0.66 \cdot 70 = \boxed{53}
\end{aligned}
$$

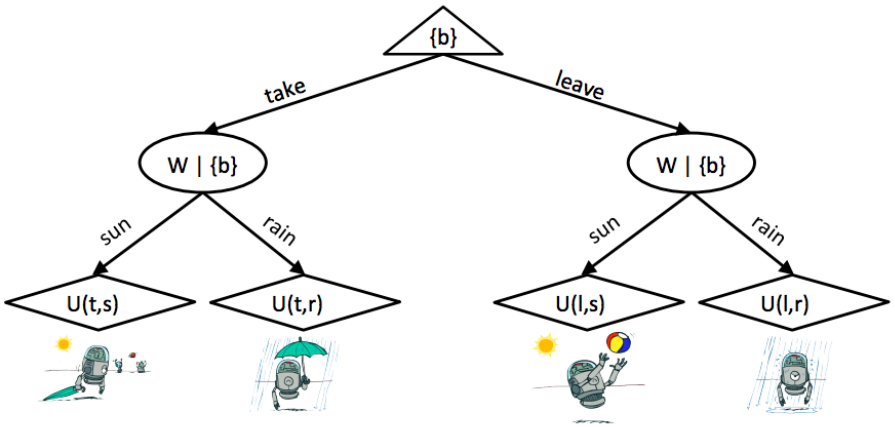
剩下要做的就是在这些计算出的效用值中取最大值来确定最大期望效用（MEU）：

$$
MEU(F = \text{bad}) = \max_a EU(a|\text{bad}) = \boxed{53}
$$

产生最大期望效用的行动被采纳，所以这就是决策网络向我们推荐的行动。更正式地说，产生最大期望效用的行动可以通过对期望效用取argmax来确定。

结果树

在本笔记开头我们提到决策网络涉及一些类似期望最大化的元素，那么让我们来讨论这到底是什么意思。我们可以将在决策网络中选择对应于使期望效用最大化的行动，拆解为一棵结果树。我们上面的天气预报示例可拆解为以下结果树：



顶部的根节点是一个最大化节点，就像在期望最大化算法中一样，并且由我们控制。我们选择一个动作，这会将我们带到树的下一层，由机会节点控制。在这一层，机会节点以与基于底层贝叶斯网络运行概率推理得出的后验概率相对应的概率解析为最终层的不同效用节点。这与普通的期望最大化算法究竟有何不同？唯一真正的区别在于，对于结果树，我们用在任何给定时刻所知道的信息（在花括号内）来注释我们的节点。

完美信息的价值

在我们到目前为止所涵盖的所有内容中，我们通常一直假设我们的智能体拥有解决特定问题所需的所有信息，并且/或者没有获取新信息的途径。在实际情况中，几乎并非如此，而决策中最重要的部分之一就是要知道收集更多证据以帮助决定采取何种行动是否值得。观察新证据几乎总是会有一些成本，无论是在时间、金钱还是其他方面。在本节中，我们将讨论一个非常重要的概念——完美信息的价值（VPI），它从数学上量化了如果智能体观察到一些新证据，其最大期望效用预计会增加的量。我们可以将学习某些新信息的VPI与观察该信息相关的成本进行比较，以决定观察是否值得。

通用公式

与其简单地给出计算新证据的完美信息价值的公式，不如让我们来进行一个直观的推导。从我们上面的定义可知，完美信息的价值是指如果我们决定观察新证据，我们的最大期望效用预计会增加的量。我们根据当前证据 e 知道我们当前的最大效用：

$$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$$

此外，我们知道如果在行动之前观察到一些新证据 e' ，那么此时我们行动的最大期望效用将变为

$$MEU(e,e') = \max_a \sum_s P(s|e,e')U(s,a)$$

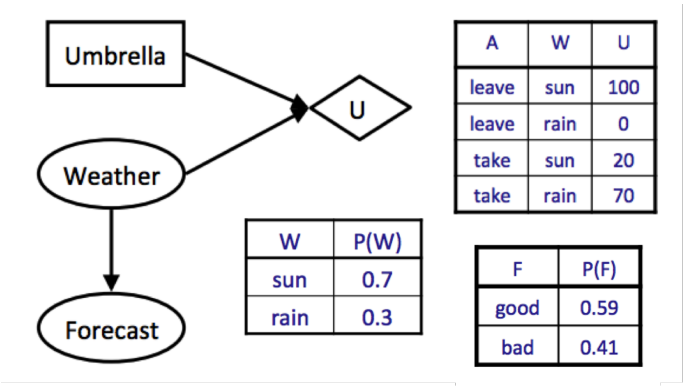
然而，请注意我们并不知道会得到什么新证据。例如，如果我们事先不知道天气预报并选择去观察它，我们观察到的预报可能是好的也可能是坏的。因为我们不知道会得到什么新证据 e' ，所以我们必须将其表示为一个随机变量 E' 。如果我们不知道从观察中获得的证据会告诉我们什么，那么当我们选择观察一个新变量时，我们如何表示将会得到的新的最大期望效用呢？答案是计算最大期望效用的期望值，虽然这有点拗口，但这是自然的做法：

$$MEU(e,E') = \sum_{e'} P(e'|e)MEU(e,e')$$

观察一个新的证据变量会产生一个不同的最大期望效用（MEU），其概率对应于观察证据变量每个值的概率。因此，通过如上计算 $MEU\left(e, E'\right)$ ，我们可以计算出如果选择观察新证据，我们期望的新MEU会是多少。我们现在差不多完成了——回到我们对信息价值（VPI）的定义，我们想找出如果选择观察新证据，我们的MEU预期会增加多少。我们知道当前的MEU以及如果选择观察时新MEU的期望值，所以预期的MEU增加量就是这两个值的差！实际上，

$$\boxed{VPI(E'|e) = MEU(e,E') - MEU(e)}$$

在这里，我们可以将 $VPI\left(E' \mid e\right)$ 理解为 “在我们当前的证据 e 下，观察新证据 E 的值”。让我们最后一次回顾天气场景来逐步分析一个例子：



如果我们没有观察到任何证据，那么我们的最大期望效用可以如下计算：

$$\begin{aligned} MEU(\varnothing) &= \max_a EU(a) \\ &= \max_a \sum_w P(w)U(a,w) \\ &= \max\{0.7\cdot 100 + 0.3\cdot 0, 0.7\cdot 20 + 0.3\cdot 70\} \\ &= \max\{70, 35\} \\ &= 70 \end{aligned}$$

注意，当我们没有证据时的惯例是写成 $MEU(\varnothing)$，表示我们的证据是空集。现在假设我们正在决定是否观察天气预报。我们已经计算出 $\text{MEU}(F = \text{bad}) = 53$，并且假设对 $F =$ 进行相同的计算得出 $MEU(F = \text{good}) = 95$。我们现在准备计算 $MEU(e, E')$：

$$\begin{aligned} MEU(e,E') &= MEU(F) \\ &= \sum_{e'} P(e'|e) MEU(e,e') \\ &= \sum_f P(F=f) MEU(F=f) \\ &= P(F=\text{good}) MEU(F=\text{good}) + P(F=\text{bad}) MEU(F=\text{bad}) \\ &= 0.59\cdot 95 + 0.41\cdot 53 \\ &= 77.78 \end{aligned}$$

因此我们得出 $\text{VPI}(F) = \text{MEU}(F) - \text{MEU}(\varnothing) = 77.78 - 70 = \boxed{7.78}$。

价值信息的属性

完全信息的价值具有几个非常重要的属性，即：

- 非负性。$\forall E', eVPI(E' \mid e) \geq 0$

  观察新信息总能让你做出更明智的决策，因此你的最大期望效用只会增加（或者如果该信息与你必须做出的决策无关，则保持不变）。

- 非可加性。$\text{VPI}(E_j, E_k \mid e) \neq \text{VPI}(E_j \mid e) + \text{VPI}(E_k \mid e)$ 一般情况下。

  这可能是三个属性中最难凭直觉理解的。这是正确的，因为一般来说，观察到一些新证据 E_j 可能会改变我们对 E_k 的关注程度；因此，我们不能简单地将观察 E_j 的VPI与观察 E_k 的VPI相加，来得到观察两者的VPI。

相反，观察两个新证据变量的VPI等同于观察其中一个，将其纳入我们当前的证据中，然后再观察另一个。这由VPI的顺序独立性属性所概括，下文将对此进行更多描述。

•顺序独立性。

$$\mathrm{VPI}(E_j, E_k \mid e) = \mathrm{VPI}(E_j \mid e) + \mathrm{VPI}(E_k \mid e, E_j) = \mathrm{VPI}(E_k \mid e) + \mathrm{VPI}(E_j \mid e, E_k)$$

观察多个新证据，无论观察顺序如何，在最大期望效用上产生的增益相同。这应该是一个相当直观的假设——因为在观察任何新证据变量之前我们实际上不会采取任何行动，所以我们是一起观察新证据变量还是按照任意顺序依次观察其实并无实际影响。