

Then we turn our attention to another kind of rules. We give a simple example. Suppose an online vendor wants to give a special discount if it is a customer's birthday. An easy way to represent this business policy with rules is as follows:

$R1$: If birthday, then special discount.

$R2$: If not birthday, then not special discount.

This solution works properly in cases where the birthday is known. But imagine a customer who refuses to provide his birthday because of privacy concerns. In such a case, the preceding rules cannot be applied because their premises are not known. To capture this situation we need to write something like

$R1$: If birthday, then special discount.

$R2'$: If birthday is not known, then not special discount.

However, the premise of rule $R2'$ is not within the expressive power of predicate logic. Thus we need a new kind of rule system. We note that the solution with rules $R1$ and $R2$ works in cases where we have complete information about the situation (for example, either birthday or not birthday). The new kind of rule system will find application in cases where the available information is incomplete.

Predicate logic and its special cases are *monotonic* in the following sense. If a conclusion can be drawn, it remains valid even if new knowledge becomes available. But if rule $R2'$ is applied to derive "not special discount," then this conclusion may become invalid if the customer's birthday becomes known at a later stage and it happens to coincide with the purchase date. This behaviour is *nonmonotonic* because the addition of new information leads to a loss of a consequence. Thus we talk of nonmonotonic rules to distinguish them from monotonic rules (which are a special case of predicate logic). In this chapter we discuss both monotonic and nonmonotonic rules.