Lógica proposicional: dedução natural

Tiago Massoni
Lógica para Computação
Universidade Federal de Campina Grande

Já repeti algumas vezes, mas vou voltar novamente a uma mesma ideia: *verificar a validade de um argumento* é o que interessa em lógica. Tudo que você viu e vai ver gira em torno desta mesma tarefa. Como exemplo, consideremos alguns argumentos para refletir sobre isso na Tabela 1.

Esses argumentos todos parecem fazer sentido, não acha? Na real, parecem simples jogos de palavras que em essência têm um significado trivial. Se codificarmos o Argumento A, por exemplo, chegaremos a algo do tipo:

- Primeira premissa: $d \land c$
- Segunda premissa: p
- Conclusão: $c \land p$

Exercício. Codifique argumentos B e C da mesma forma.

Você sabe como escrever estas fórmulas usando um formato de argumento, colocando as premissas do lado esquerdo e a conclusão do lado direito de algum símbolo de verificação. Por ora, usamos apenas o símbolo da vinculação semântica ($\models$), que tem a função de definir a

Table 1: Exemplo de Argumentos

Argumento A	O programa calcula o número de disciplinas cursadas e calcula o CRE do aluno. O programa foi feito em Python. Portanto, o programa calcula o CRE do aluno e foi feito em Python.
Argumento B	O programa foi feito em Python. Não é verdade que o programa não calcula o CRE e o número de disciplinas cursadas do aluno. Portanto, não é verdade que o programa não foi feito em Python, e o programa calcula o CRE do aluno.
Argumento C	Se é feito o acesso a um índice inválido de um array, se ocorrer o lançamento de NullPointerException, o programa quebra. Se há o acesso a um índice inválido, ocorre o lançamento de NullPointerException. Ocorre o acesso a índice inválido de um array. Portanto, o programa quebra.

validade semântica do argumento.

$$d \wedge c, p \models c \wedge p \tag{1}$$

Você sabe também o que significa este símbolo, suponho; quando o valor das premissas for verdadeiro, *para todas elas*, o valor da conclusão deve ser verdadeiro também. Podemos utilizar uma tabela verdade como algoritmo para testar a validade de um argumento: basta explorar todas as interpretações para as premissas e para a conclusão, e se for encontrada uma delas que possui premissas verdadeiras e conclusão falsa, o argumento é inválido. Se, depois de checar todas as linhas, nenhum caso desses for observado, o argumento é considerado válido (ou seja, observa-se a vinculação).

Se você se lembra bem, este algoritmo, apesar de correto e útil para poucas proposições atômicas, torna-se intratável para muitos átomos. Isso acontece porque a tabela para uma fórmula que possui n átomos é composta por 2^n linhas – a construção desta tabela possui custo polinomial. Assim, teremos problemas para verificar argumentos em geral automaticamente, desta forma.

1 Dedução Natural

Por outro lado, podemos não usar um algoritmo para fazer esta verificação. Com a intervenção humana, podemos verificar um argumento por *demonstração*, muitas vezes de forma mais simples. Como provador, seu objetivo é transformar as premissas na conclusão, através de *transformações sintáticas* corretas. Essas transformações são corretas se seguirem a aplicação de *regras de inferência*, que garantem a consistência das fórmulas resultantes da transformação de verdades anteriores. Ao aplicar essas regras às premissas, esperamos que chegaremos a mais algumas fórmulas, e se aplicarmos mais regras a estas, em algum momento chegaremos na fórmula da conclusão. Este processo é codificado no argumento da seguinte forma:

$$\phi_1, \phi_2, ..., \phi_n \vdash \psi \tag{2}$$

Esta expressão[1] tem o nome técnico de *sequent*; ela é válida se uma prova pode ser encontrada.

Um dos métodos usados para provar um argumento – e que você vai praticar aqui – é chamado de *dedução natural*; deduzir requer um pouco de criatividade, talvez lembrando um pouco de programação, já que demanda você pensar numa *estratégia* para a aplicação das regras

[1] Matemáticos e lógicos adoram letras gregas, mas não se incomode muito com isso. A ideia aqui é que uma letra grega representa uma fórmula de qualquer tamanho em lógica proposicional.

de inferência. Às vezes não é muito óbvio definir qual regra aplicar em cada caso, por isso quero que você aprenda a técnica de dedução com *estratégias* gerais, não que você decore as regras. Aprendendo estratégias de dedução, as regras surgirão naturalmente[2] no processo.

2 Regras de inferência

Para que estas estratégias possam ser colocadas em prática, precisamos de "ferramentas". Um eletricista sabe fazer o conserto de uma fiação, mas sem ferramentas ele não o faz – é o mesmo para nós. Se você pensar que transformar as fórmulas é o seu trabalho de "conserto", precisamos das ferramentas mais apropriadas; nossas ferramentas são as *regras de inferência*. Enquanto te mostro as ferramentas (regras), uma a uma, considere que você está se equipando; gosto de dizer que adquirimos uma caixa de ferramentas (conjunto de regras), que estará com você enquanto estiver realizando dedução natural.

As regras são organizadas por conectivo, divididas entre regras de *introdução* e regras de *eliminação*. Vamos iniciar com as duas regras referentes à conjunção:

$$\wedge_i \quad \frac{\phi \quad \psi}{\phi \wedge \psi}$$

$$\wedge_e \quad \frac{\phi \wedge \psi}{\phi}$$

Uma regra deve ser lida de cima para baixo; acima do traço, temos as fórmulas de entrada da regra, necessariamente já conhecidas como verdadeiras, e abaixo do traço, temos a fórmula resultante da regra. Na regra de introdução da conjunção ($\wedge_i$), se forem provadas como verdadeiras as fórmulas ϕ e ψ, podemos deduzir a conjunção entre os dois. Já na eliminação da conjunção ($\wedge_e$), basta que se tenha como verdadeira uma conjunção de duas fórmulas ($\phi \wedge \psi$), qualquer uma delas pode ser tomada como verdade, já que as duas são verdadeiras ao mesmo tempo.

Você deve estar achando essas regras bem fáceis, e nisso concordo com você. Ainda assim, saber aplicá-las exige um pouco mais de habilidade, e é neste ponto que eu quero exercitar a habilidade de dedução. Vamos voltar ao Argumento (1), reescrevendo-o para a utilização da dedução natural.

$$d \wedge c, p \vdash c \wedge p \tag{3}$$

Supondo que temos que realizar a prova de validade deste argumento, vamos montar o esquema de dedução, que é o primeiro passo

[2] Perdão pelo péssimo trocadilho.

para uma prova correta e organizada. Isso é importante, pois demonstrações são feitas para convencer outras pessoas, por isso têm que ser precisas e claras. A primeira coisa é copiar as premissas, uma por linha numerada.

1.	$d \wedge c$	premissa
2.	p	premissa

A ideia aqui é usar estas duas linhas como origem para transformações que vão gerar a conclusão. Acho importante você deixar um espaço e já *escrever a conclusão*, na última linha, mesmo sem saber ainda qual o número desta linha na prova completa.

1.	$d \wedge c$	premissa
2.	p	premissa
3.		
4.		
5.		
6.	$c \wedge p$	

Essa prática ajuda a saber onde queremos chegar, facilitando a definição da estratégia a utilizar. Olhando para a conclusão, você deve se perguntar: "Qual o caminho mais apropriado para chegar a este tipo de conclusão?" O que chamo de tipo de conclusão consiste na estrutura da fórmula. Neste caso, a conclusão tem uma estrutura de conjunção – este é o conectivo principal da fórmula. Outros exemplos:

- $\neg(p \vee (q \wedge r))$: estrutura de negação;
- $(p \vee q) \rightarrow (p \wedge r)$: estrutura de implicação;
- $(p \wedge q) \vee r$: estrutura de disjunção.

Voltando à nossa dedução, qual seria a melhor forma de chegar a uma conjunção? Que regras deveríamos aplicar para chegar lá? Como só vimos duas regras até agora, é simples: para chegar a esta conjunção ($c \wedge p$), precisamos chegar a cada um dos lados da conjunção *isoladamente*, para no final juntar os dois no resultado, utilizando a regra $\wedge_i$. Você poderia já colocar estes dois lados *nas linhas acima* da conclusão, assim:

1.	$d \wedge c$	premissa
2.	p	premissa
3.		
4.	c	
5.	p	
6.	$c \wedge p$	

Agora nosso trabalho se resume a tentar deduzir c e p isoladamente. A proposição p já é premissa, então ela já é válida nesta dedução, por isso vou apagar.

1.	$d \wedge c$	premissa
2.	p	premissa
3.		
4.	c	
5.	$c \wedge p$	

O problema que resta é chegar ao c isolado. Você tem agora que olhar as premissas para tentar delas derivar o resultado desejado. Veja que na Linha 1 temos a conjunção cujo um dos lados é exatamente c, que eu posso extrair usando a regra $\wedge_e$. A criação da nova linha exige uma justificativa, que fica do lado direito, incluindo o nome da regra e a linha de onde vêm as entradas da regra (nesta regra, só há uma entrada, a Linha 1):

1.	$d \wedge c$	premissa
2.	p	premissa
3.	c	$\wedge$e, 1
4.	$c \wedge p$	

Temos os dois lados isolados na dedução, assim você pode finalizar a prova com o último passo de introduzir a conjunção. Como a linha já está colocada, falta apenas justificá-la como resultado.

1.	$d \wedge c$	premissa
2.	p	premissa
3.	c	$\wedge$e, 1
4.	$c \wedge p$	$\wedge$i, 2,3

Perceba que duas linhas foram usadas na justificativa, já que esta regra precisa de duas entradas (cada lado da conjunção).

Exercício. Verifique a validade do argumento usando dedução natural: $p \wedge q, r \vdash q \wedge r$.

Adicionemos então mais algumas regras na nossa caixa de ferramenta. Essas têm a ver com dupla negação:

$$\neg\neg_e \quad \frac{\neg\neg\phi}{\phi}$$

$$\neg\neg_i \quad \frac{\phi}{\neg\neg\phi}$$

Lembre-se do Argumento B da Tabela 1; se codificarmos a linguagem natural deste argumento, teremos o seguinte sequent:

$$p, \neg\neg(c \wedge d) \vdash \neg\neg p \wedge c \tag{4}$$

Esta dedução, a partir das duas premissas, exige a introdução e a eliminação de duplas negações, além da aplicação de regras de conjunção.

1.	p	premissa
2.	$\neg\neg(c \wedge d)$	premissa
3.	$\neg\neg p$	$\neg\neg$i 1
4.	$c \wedge d$	$\neg\neg$e, 2
5.	c	$\wedge$e 4
6.	$\neg\neg p \wedge c$	$\wedge$i 3,5

Exercício. Verifique a validade do argumento usando dedução natural: *O programa calcula o número de disciplinas cursadas e o CRE do aluno, e foi feito em Python. O programa imprime o histórico na tela e em folha de papel. Portanto, o programa calcula o CRE do aluno e também imprime o histórico na tela.*

Vamos começar a manipular agora o conectivo de *implicação*. Começamos do jeito mais simples, a sua eliminação (deixemos a introdução para depois, já que precisaremos de um conceito novo). A eliminação da implicação é definida assim:

$$\rightarrow_e \quad \frac{\phi \quad \phi \rightarrow \psi}{\psi}$$

Parte dos silogismos de Aristóteles lembra esta regra, que ficou conhecido na história do estudo da lógica como *Modus Ponens* (Modo de Afirmar). Nela, se sabemos ser verdade a própria implicação, e sabemos ser verdade seu lado esquerdo, com certeza o lado direito é verdadeiro. Tomemos como exemplo p: O programa recebeu como entrada um número inteiro, e $p \rightarrow q$: se o programa recebe como parâmetro um número inteiro, então o programa retorna um valor booleano. Se ambas as fórmulas são verdadeiras, com certeza poderemos afirmar que o programa *retorna um valor booleano*, já que a condição suficiente para isso aconteceu (p). Argumento C codificado será nosso exemplo de dedução:

$$i \rightarrow (n \rightarrow q), i \rightarrow n, i \vdash q \tag{5}$$

A dedução pode ser feita apenas aplicando a regra $\rightarrow_e$ três vezes, a partir do fato de i ser uma premissa, o que inicia as deduções permitidas para chegarmos à conclusão. Preste atenção nas justificativas na coluna da direita, com seus números de linha para cada aplicação.

1.	$i \to (n \to q)$	premissa
2.	$i \to n$	premissa
3.	i	premissa
4.	$n \to q$	$\to$e 1,3
5.	n	$\to$e 2,3
6.	q	$\to$e 4,5

Uma outra regra que vamos adicionar é análoga a esta: *Modus Tolens* (Modo de Negar), que permite deduzir a fórmula do lado esquerdo de uma implicação quando tem-se como verdadeira a *negação da fórmula do lado direito.*

$$\text{MT} \quad \frac{\phi \to \psi \quad \neg\psi}{\neg\phi}$$

O *sequent* abaixo pode ser deduzido rapidamente com esta regra:

$$a \to (i \to n), a, \neg n \vdash \neg i \tag{6}$$

1.	$a \to (i \to n)$	premissa
2.	a	premissa
3.	$\neg n$	premissa
4.	$i \to n$	$\to$e 1,2
5.	$\neg q$	MT 4,5

3 Hipóteses e Caixas Temporárias

Aqui tem uma novidade: teremos agora como *introduzir* uma implicação onde não existe uma. Lembre-se de que uma fórmula como $p \to q$ não afirma p nem q; não sabemos se a entrada do programa é um número inteiro, ou se ele retornou um valor booleano. Sabemos apenas que *se a entrada for inteiro, a saída será booleano*, só isso. Criar uma implicação em uma dedução exige que criemos *hipóteses temporárias*, cujo efeito (escopo) é limitado a apenas um pedaço da dedução – um trecho dentro da lista ordenada de linhas numeradas. Este trecho temporário é representado por uma espécie de quadro de contorno sólido, que chamamos de *caixa*.

Que tal um exemplo antes de formalizar a regra?

$$\neg j \to \neg c \vdash c \to \neg\neg j \tag{7}$$

Vou começar escrevendo a premissa, seguida da conclusão no final. O espaço em branco entre elas será nossa área de trabalho, onde vamos pensar a estratégia.

1.	$\neg j \rightarrow \neg c$	premissa
2.		
3.		
4.		
5.	$c \rightarrow \neg\neg j$	

Nosso objetivo é chegar numa implicação, mas ela não existe na dedução ainda, vamos assim precisar criar de algum lugar. Vamos criar as condições para que uma implicação possa ser deduzida: vou *assumir como verdadeira* a fórmula esquerda da implicação desejada (c). Não quer dizer que ela seja, talvez não seja, mas vou assumir como uma hipótese; já que não sabemos se ela existe, então a colocaremos dentro de um espaço temporário, onde as fórmulas anteriores também são verdadeiras, junto com essa hipótese.

1.	$\neg j \rightarrow \neg c$	premissa
2.	c	hipótese
3.	...	...
4.	$c \rightarrow \neg\neg j$	

É importante pensar que a caixa é uma sub-dedução dentro da dedução maior; neste caso, podemos usar Linhas 1 e 2 como verdade para a dedução dentro da caixa. Neste caso, se conseguirmos deduzir, no final dela, o lado direito da implicação desejada ($\neg\neg j$), chegamos à conclusão que queríamos; partimos, assumindo o lado direito, numa dedução que leva ao lado direito, ou seja, *o lado direito foi realmente implicado a partir do lado esquerdo*!

1.	$\neg j \rightarrow \neg c$	premissa
2.	c	hipótese
3.	...	...
4.	...	...
5.	$\neg\neg j$	...
6.	$c \rightarrow \neg\neg j$	

Perceba que nosso problema da caixa para fora está resolvido, basta agora resolver o miolo da caixa. Precisamos então, a partir das Linhas 1 e 2, chegar em $\neg\neg j$. Mas, veja, o final da caixa, onde queremos chegar, é justamente o lado esquerdo negado da implicação da Linha 1! Se eu perguntar qual a estratégia para chegar na negação do lado esquerdo de uma implicação, você vai rapidamente me responder que precisamos usar Modus Tolens, de algum jeito. Para isso, precisamos negar duplamente c.

1.	$\neg j \to \neg c$	premissa
2.	c	hipótese
3.	$\neg\neg c$	¬¬i 2
4.	$\neg\neg j$	MT 3,1
5.	$c \to \neg\neg j$	→i 2-4

A justificativa da aplicação desta regra tem que incluir os números de linha da caixa (2–4). Segue a formalização da regra:

$$\to\mathrm{i}\quad \frac{\boxed{\begin{array}{c}\phi\\ \vdots\\ \psi\end{array}}}{\phi \to \psi}$$

Você já tem ferramentas para manipular conjunções, negações duplas e implicações; vamos incluir disjunções nesse conjunto. Para introduzi-las, não há segredo, basta adicionar uma disjunção a qualquer fórmula que já tenha sido deduzida.

$$\vee\mathrm{i}\quad \frac{\phi}{\psi \vee \phi}$$

Já a eliminação exige um pouco mais da sua atenção. Considere duas proposições, eu fui à pé ao trabalho ($onfoot$) e eu fui de carro para o trabalho (car), e considere que você deduziu a expressão $onfoot \vee car$. Sabemos que uma das duas parcelas é verdadeira, ou talvez ambas, mas não temos ideia qual. Mas você vai concordar que, independente de como eu fui, eu *cheguei ao trabalho*, que é uma proposição distinta das duas parcelas da disjunção. Existem então certas proposições (chamamos de X) que podem ser deduzidas de uma disjunção; precisamos verificar se ela se confirma *para qualquer uma das parcelas da disjunção*, assim vamos usar caixas para assumir os dois lados como hipótese, em paralelo, de forma independente.

$$\vee\mathrm{e}\quad \frac{\phi \vee \psi \qquad \boxed{\begin{array}{c}\phi\\ \vdots\\ X\end{array}} \quad \boxed{\begin{array}{c}\psi\\ \vdots\\ X\end{array}}}{X}$$

Vejamos um exemplo de argumento em que esta regra é usada.

$$a \to d \vdash (z \vee a) \to (z \vee d) \tag{8}$$

Voltarei a falar de estrategia, por isso quero montar a estrutura da dedução antes de partir para a aplicação de regras. Por isso coloco logo a conclusão no final.

1.	$a \to d$	premissa
2.	...	
3.	...	
4.	$(z \vee a) \to (z \vee d)$	

Perceba que nosso alvo é uma *implicação*, por isso podemos tentar uma estratégia de criar esta implicação. Prossigo criando a caixa então e definindo seus limites.

1.	$a \to d$	premissa
2.	$(z \vee a)$	hipótese
3.	...	
4.	$(z \vee d)$	
5.	$(z \vee a) \to (z \vee d)$	

Como então chegar, dentro da caixa, em $(z \vee d)$? Precisamos de z ou d para criar esta disjunção, mas não temos nenhum; não posso remover z da disjunção simplesmente, ou remover d da implicação. Mas, temos o que eu chamo de "presente": uma disjunção que pode ser eliminada ($z \vee d$)! Com ela temos uma estratégia sem pensar muito, já que podemos tentar a eliminação assumindo como hipótese cada uma de suas parcelas em caixas separadas, e tentar chegar no mesmo resultado (neste caso, $z \vee d$, que é a conclusão da caixa).

1.	$a \to d$	premissa
2.	$(z \vee a)$	hipótese
3.	z	hipótese
4.	...	
5.	$(z \vee d)$	
6.	a	hipótese
7.	...	
8.	$(z \vee d)$	
9.	$(z \vee d)$	
10.	$(z \vee a) \to (z \vee d)$	

Na primeira caixa interna, z transforma-se em apenas um passo na fórmula $z \vee d$, pela introdução da disjunção. Já na segunda, a pode ser usado para eliminar a implicação da Linha 1, chegando a d, que pode ser enriquecido com uma disjunção da mesma forma, chegando a $z \vee d$. Preste atenção nas justificativas; a regra de eliminação de disjunção é aplicada na Linha 8, tendo como entrada a disjunção e as duas caixas usadas.

1.	$a \to d$	premissa
2.	$(z \vee a)$	hipótese
3.	z	hipótese
4.	$(z \vee d)$	$\vee$i 2
5.	a	hipótese
6.	d	$\to$e 5,1
7.	$(z \vee d)$	$\vee$i 6
8.	$(z \vee d)$	$\vee$e 2,3-4,5-7
9.	$(z \vee a) \to (z \vee d)$	$\to$i 2-8

Exercício. Verifique a validade do argumento usando dedução natural: $(p \vee q) \vee r \vdash p \vee (q \vee r)$.

4 Contradição

> Ninguém pode dizer de algo que este é e que este não é em relação à mesma coisa, e ao mesmo tempo. (Aristóteles)

Vamos aumentar nossa caixa de ferramenta para lidar com negações. Não estou falando de negações duplas, que você pode manipular facilmente. Agora, a questão é como deduzir fórmulas negadas, e como manipulá-las. O conceito mais importante aqui é a *contradição lógica*, que, mesmo sendo um absurdo intuitivamente – já que não faz sentido uma proposição ser verdadeira e falsa ao mesmo tempo – tem grande utilidade para o raciocínio envolvido na dedução natural.

Antes de qualquer coisa, a contradição possui uma definição: ela caracteriza uma situação em que algo afirmado é também negado, ou se preferir, uma conjunção de contrários. Isso é estabelecido pela seguinte regra de inferência, que chamamos de *eliminação da negação*. O símbolo que usaremos para representar uma contradição será $\perp$.

$$\neg\text{e} \quad \frac{\phi \quad \neg\phi}{\perp}$$

A propriedade mais importante da contradição é sua capacidade (ou bizarrice, diriam alguns) de, sendo premissa, *permiti deduzir qualquer fórmula*. Em um argumento, havendo uma premissa como contradição, qualquer outra conclusão é válida, pela própria característica da vinculação semântica. Como você deve se lembrar, um argumento só é inválido quando suas premissas forem verdadeiras e sua conclusão for falsa. Com uma só contradição do lado esquerdo do *sequent*, não há como ter premissas verdadeiras, por isso o argumento se mantém válido. Esta propriedade é definida como regra, a introdução da contradição:

$$\perp\mathrm{i}\quad \frac{\perp}{\phi}$$

Um exemplo do uso destas duas regras acontece no seguinte argumento: $\neg c \vee t \vdash c \rightarrow t$. Pela propriedade geral dos argumentos com implicações na conclusão, vou simplificar este argumento para $\neg c \vee t, c \vdash t$. Como uma das premissas é uma disjunção, é uma boa tentar usar a estratégia que vem de graça com ela.

1.	$\neg c \vee t$	premissa
2.	c	premissa
3.	$\neg c$	hipótese
4.	...	
5.	t	
6.	t	hipótese
7.	t	

A segunda caixa já está finalizada: a hipótese já é o resultado que precisamos. Na primeira, precisamos, a partir de $\neg c$, chegar a t, que não pode ser removido da disjunção da Linha 1. Mas temos um trunfo: dentro desta caixa tanto c quanto $\neg c$ são verdades, o que nos leva a descobrir que dentro desta caixa o sistema lógico se degenerou, já que provamos nela uma contradição. Assim você pode aplicar as duas regras que acabamos de ver, uma para colocar a contradição e outra para deduzir uma fórmula qualquer.

1.	$\neg c \vee t$	premissa
2.	c	premissa
3.	$\neg c$	hipótese
4.	$\perp$	$\neg$e 2,3
5.	t	$\perp$i 4
6.	t	hipótese
7.	t	$\vee$e 1,3-5,6

A essas duas regras, incluímos uma regra mais útil e relevante: sempre que tivermos que deduzir uma fórmula negada, podemos assumir sua afirmação como hipótese em uma caixa, e tentar chegar na contradição. Isso é suficiente para afirmar que a fórmula deve ser negada. Nomeamos esta regra como introdução da negação.

$$\neg\mathrm{i}\quad \frac{\boxed{\begin{array}{c}\phi\\ \vdots\\ \perp\end{array}}}{\neg\phi}$$

Outra regra análoga, para simplificar sua vida: Redução Ao Absurdo (RAA), que simplesmente inverte a hipótese. Assim assume-se a negação da fórmula em que queremos chegar, tentando deduzir a contradição.

$$\text{RAA}\ \frac{\boxed{\begin{array}{c}\neg\phi\\ \vdots\\ \bot\end{array}}}{\phi}$$

Por exemplo, o mesmo argumento já provado anteriormente com Modus Tolens pode ser deduzido de forma diferente:

$$a \to (i \to n), a, \neg n \vdash \neg i \tag{9}$$

Além das premissas, você deve esboçar a conclusão; neste caso, queremos chegar em $\neg i$.

1.	$a \to (i \to n)$	premissa
2.	a	premissa
3.	$\neg n$	premissa
4.	$\vdots$	
5.	$\neg i$	

Queremos chegar em uma fórmula *negada*, o que nos leva a uma estratégia de introduzir esta negação. Pela regra, podemos assumir a afirmação desta fórmula, e tentar chegar na contradição.

1.	$a \to (i \to n)$	premissa
2.	a	premissa
3.	$\neg n$	premissa
4.	i	hipótese
5.	$\vdots$	
6.	$\bot$	
7.	$\neg i$	

Para chegar nesta contradição precisamos deduzir algum oposto às fórmulas que já assumimos como verdadeiras. Desta forma, podemos deduzir n, que entra em contradição com $\neg n$, umas das premissas.

1.	$a \to (i \to n)$	premissa
2.	a	premissa
3.	$\neg n$	premissa
4.	i	hipótese
5.	$i \to n$	$\to$e 1,2
6.	n	$\to$e 4,5
7.	$\perp$	$\neg$e 3,6
8.	$\neg i$	$\neg$i 4-7

5 Regra do intermediário excluído

Para finalizar, nossa caixa de ferramentas se torna completa com uma regra distinta das outras. Sua diferença marcante é que ela *não exige nenhuma linha de entrada*, ou seja, pode ser aplicada de forma livre, em qualquer ponto de uma dedução. Esta regra parte da suposição de que qualquer proposição em disjunção com sua negação forma uma fórmula válida.

$$\text{LEM} \quad \overline{\phi \lor \neg\phi}$$

Fica interessante adicionar uma disjunção como esta em deduções em cujas estratégias tradicionais não funcionam; lembre-se de que a disjunção sempre traz uma estratégia gratuita. Vejamos o caso do argumento:

$$m \to d \vdash \neg m \lor d \tag{10}$$

Ao tentar deduzir uma disjunção, uma estratégia convencional pode ser tentar chegar a $\neg m$ ou d isoladamente, e depois adicionar a disjunção. Mas neste caso, não conseguimos deduzir nenhum dos dois sozinhos. Por este motivo, adiciono uma disjunção do nada, envolvendo m, que me será útil como estratégia.

1.	$m \to d$	premissa
2.	$m \lor \neg m$	LEM
3.	m	hipótese
4.	$\vdots$	
5.	$\neg m \lor d$	
6.	$\neg m$	hipótese
7.	$\vdots$	
8.	$\neg m \lor d$	
9.	$\neg m \lor d$	

Estratégia? Tentar chegar na conclusão a partir desta disjunção que acabei de colocar. Acabou funcionando, porque posso chegar em $\neg m \vee d$ em ambas as caixas, eliminando implicações ou apenas adicionando disjunções.

1.	$m \to d$	premissa
2.	$m \vee \neg m$	LEM
3.	m	hipótese
4.	d	$\to$e 3,1
5.	$\neg m \vee d$	$\vee$i 4
6.	$\neg m$	hipótese
7.	$\neg m \vee d$	$\vee$i 6
8.	$\neg m \vee d$	$\vee$e 2,3-5,6-7