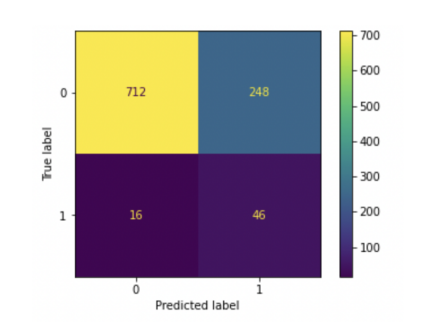
**Matriz de confusão**



Imaginando que seja um modelo que prevê fraudes, podemos entender que quando acontece uma fraude, o modelo apenas erra em 16 casos. Mas quando não há uma frande, o modelo erra em 248, ou seja, o modelo está prevendo muitas classes como 1, mesmo que erroneamente, podemos perceber isso ao observar que ele está acertando muito pouco quando a prevê a classe 1, sempre que ele prevê 1, tem uma probabilidade de 16% dele estar certo(precisão). Nesse caso em questão não haveria muitos problemas, uma vez que é mais importante diminuir a taxa de erro quando o cliente realmente vai cometer uma fraude, mesmo que para isso aconteça uma diminuição no precision, pois o modelo está prevendo classes demais como 1, e por consequência, errando algumas delas.

Mas em exemplos como se ao invés de fraude fosse inadimplência e que quiséssemos dar um desconto para as pessoas menos propensas a pagar a dívida, é mais importante estar preciso quando o modelo prevê 1, ou seja, uma boa precisão. Nesse caso que ele está prevendo muitos valores como 1 para diminuir o erro quando o true label for 1, seria péssimo, pois ele está prevendo muitos registros como 1, apesar de ser 0, ou seja, estaríamos dando desconto para pessoas que não precisam.

Ou seja, bem resumidamente, quando queremos um modelo que acerte muito quando o true label for 1, usaremos o **recall** como métrica. Por outro lado, quando queremos que o modelo seja preciso quando prever algo como 1, usaremos **precision**.

* **Precision** = TP / (TP / FP): De todos que o algoritmo previu positivo, quantos realmente são?
* **Recall** = TP / (TP / FN): De todos que realmente são positivos, quais o algoritmo previu corretamente?

Em linhas gerais, pode-se dizer que quando aumentamos um, o outro diminui, então precisa-se de uma análise minuciosa sobre como achar um meio termo entre eles para um modelo de negócios específico.

**Custo X Benefícios do modelo**

Primeiramente é importante deixar bastante claro que para subir um modelo, é necessário que o seu retorno supere os seus custos totais, ou seja, o que ele faz a empresa ganhar(informações, estratégias de otimização) seja superior ao que a empresa vai gastar(custo de máquina para rodar, custo da nuvem, se o tempo da predição afeta na qualidade do atendimento).

Um exemplo de custo Benefício que precisa ser analisado: Vamos supor que temos um modelo que queira prever se um cliente vai ou não dar Churn em uma empresa, e para aqueles que vão dar churn daremos um desconto, e o modelo chegou ao seguinte resultado:

1. Verdadeiro Positivo: 150
2. Verdadeiro Negativo: 500
3. Falso Positivo: 60
4. Falso Negativo: 40

Com isso, percebemos que 150 clientes vão receber um desconto corretamente, 40 clientes vão dar Churn sem ganhar desconto, e 60 vão ganhar desconto mesmo sem darem churn. Para calcular se vale a pena, precisamos ver se os 150 clientes que ganharam desconto vão gerar um lucro para a empresa a ponto de compensar o desconto dado incorretamente, logo, precisamos ver o LTV(lifetime value) dos clientes que iam sair, mas ficaram por causa do desconto, e observar se cobre as despesas dos descontos dado erroneamente. Além disso, precisa-se levar em conta que o modelo deixou 40 clientes saírem sem o desconto.

Outro exemplo que nos permite explorar bem a relação custo benefício é na construção de um modelo que vai analisar de uma pessoa deve ou não ser presa, tendo os seguintes resultados:

1. Verdadeiro Positivo: 200
2. Verdadeiro Negativo: 400
3. Falso Positivo: 90
4. Falso Negativo: 60

O principal problema dessa análise, é que 90 pessoas estão sendo presas mesmo sendo inocentes, esse é o principal objetivo da mudança, fazer com que o algoritmo erre menos ao decretar o positivo, ou seja, aumentar a precision. Uma boa estratégia para aumentar a precisão do algoritmo é aumentar o nível de confiança necessário para decretar uma pessoa como sendo culpada, no caso, para a classe 1, tendo isso em mente, os novos resultados ficaram:

1. Verdadeiro Positivo: 180
2. Verdadeiro Negativo: 440
3. Falso Positivo: 20
4. Falso Negativo: 110

Com isso, percebemos que apesar de ter caído o desempenho de todas as classes menos o Falso Positivo, onde houve a melhora, essa mudança provavelmente seria válida por se tratar de um modelo que quer a todo e qualquer custo minimizar os Falsos positivo, ou seja, as pessoas que são presas sendo inocentes.

Em todos os modelos de machine learning, deve-se analisar minuciosamente o que é mais custoso e o que o cliente se beneficiaria mais com a construção desse modelo, analisar também o quão ruim é X ou Y perda, para entender totalmente qual é o modelo de negócio que você está inserido e fazer um modelo com uma métrica ideal para isso.