Поскольку в бизнесе **Что такое матрица ошибок и зачем она нужна:**

**пример расчета стоимости ошибки прогнозирования**

[**https://bigdataschool.ru/blog/machine-learning-confusion-matrix.html**](https://bigdataschool.ru/blog/machine-learning-confusion-matrix.html) **(оригинал)**

Поиск баланса между спросом и предложением напрямую конвертируется в деньги, возникает вопрос, насколько выгодно применение методов Machine Learning для решения этой задачи.

С целью сопоставления предсказаний и реальности в Data Science используется **матрица ошибок (confusion matrix)** – таблица с 4 различными комбинациями прогнозируемых и фактических значений. Прогнозируемые значения описываются как положительные и отрицательные, а фактические – как истинные и ложные.

 Вообще матрица ошибок используется для оценки точности моделей в задачах классификации. Но [прогнозирование](https://bigdataschool.ru/wiki/%D0%BF%D1%80%D0%BE%D0%B3%D0%BD%D0%BE%D0%B7%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) и распознавание образов можно рассматривать как частный случай этой проблемы, поэтому confusion matrix актуальна и для измерения точности предсказаний.

Важно, что матрица ошибок позволяет оценить эффективность прогноза не только в качественном, но и в количественном выражении, т.е. измерить стоимость ошибки в деньгах. Например, каковы будут расходы на удержание пользователя, если машинное обучение предсказало, что он перестанет приносить компании пользу?

Аналогичный вопрос по предсказанию оттока (Churn Rate) актуален и в HR-сфере для удержания ключевых сотрудников, мотивация которых снижается. Впрочем, матрица ошибок может использоваться не только в рамках применения Machine Learning. По сути, этот метод оценки стоимости прогноза является универсальным аналитическим инструментом.

|  |  |
| --- | --- |
| **Прогноз** | **Реальность** |
| **+** | **—** |
| **+** | ***True Positive (истинно-положительное решение)***: прогноз совпал с реальностью, результат положительный произошел, как и было предсказано ML-моделью | ***False Positive (ложноположительное решение)***: ошибка 1-го рода, ML-модель предсказала положительный результат, а на самом деле он отрицательный |
| **—** | ***False Negative (ложноотрицательное решение)***: ошибка 2-го рода – ML-модель предсказала отрицательный результат, но на самом деле он положительный | ***True Negative (истинно-отрицательное решение)***: результат отрицательный, ML-прогноз совпал с реальностью |



Матрица ошибок (confusion matrix)

С математической точки зрения оценить точность ML-модели можно с помощью следующих метрик **[8]**:

* **Точность**– сколько всего результатов было предсказано верно;
* **Доля ошибок**;
* **Полнота**– сколько истинных результатов было предсказано верно;
* **F-мера**, которая позволяет сравнить 2 модели, одновременно оценив полноту и точность. Здесь используется среднее гармоническое вместо среднего арифметического, сглаживая расчеты за счет исключения экстремальных значений.

В количественном выражении это будет выглядеть так:

* ***P*** – число истинных результатов, ***P = TP + FN***;
* ***N***– число ложных результатов, ***N = TN + FP***.



Метрики оценки качества прогноза: полнота, точность, F-мера

Рассмотрим матрицу ошибок на практическом примере для задачи прогнозирования спроса на скоропортящуюся продукцию, которая должна быть продана конечному пользователю в течение суток. Например, букеты цветов, продающиеся по цене ***k*** рублей при закупочной стоимости в ***p*** рублей. Предположим, с помощью Machine Learning было предложена 2 варианта:

* **Положительный прогноз (+)**, что по цене ***k***будут полностью раскуплены все цветы в количестве ***n*** букетов.
* **Отрицательный прогноз (+)**, что по цене ***k***будут полностью раскуплены не все цветы, останется ***m*** не проданных букетов.

Соответственно, матрица ошибок для этого случая будет выглядеть следующим образом:

|  |  |
| --- | --- |
| **Прогноз** | **Реальность** |
| Проданы все букеты цветов | Остались не проданные ***m***букетов |
| **+**: Проданы все ***n***букетов по ***k***рублей c ценой закупки ***p*** | ***True Positive***: прогноз совпал с реальностью, все закупленные ***n***букетов проданы **Выручка**= *n\*k***Затраты** = *n\*p***Прибыль** = *n\*(k-p)***Стоимость ошибки** = *0* | ***False Positive***: ошибка 1-го рода, ML-модель предсказала, что будет ***n***продаж, а на самом деле их было ***(n-m)***, осталось ***m***не проданных букетов, которые пропали и не вернули затраты на их покупку **Выручка**= (*n-m)\*k***Затраты**= *n\*p***Прибыль** = *n\*(k-p) – m\*k***Стоимость ошибки =***m\*p* |
| **—**: Остались не проданные ***m***букетов c ценой закупки ***p*** | ***False Negative***: ошибка 2-го рода – ML-модель предсказала, что ***n*** букетов не будет продано, поэтому закупили ***(n-m)*** букетов, но спрос был на ***n*** букетов. Эффект недополученной прибыли **Выручка**= (*n-m)\*k***Затраты**= (*n-m)\*p***Прибыль** = (*n-m)\*(k-p)***Стоимость ошибки**= *m\*k* | ***True Negative***: ML-прогноз совпал с реальностью, было раскуплено ***(n-m)*** букетов по цене ***k***, сколько и было изначально закуплено по цене ***p*** **Выручка**= (*n-m)\*k***Затраты**= (*n-m)\*p***Прибыль** = (*n-m)\*(k-p)***Стоимость ошибки =***0* |

Таким образом, с помощью confusion matrix можно измерить эффективность прогноза в денежном выражении, что весьма актуально для практического бизнес-приложения Machine Learning. Впрочем, отметим еще раз, что данный метод предварительной оценки будущих сценариев можно использовать и вне сферы Data Science, оценивая риски и перспективы в рамках классического бизнес-анализа.



Точный прогноз спроса на скоропортящиеся товары позволит избежать убытков