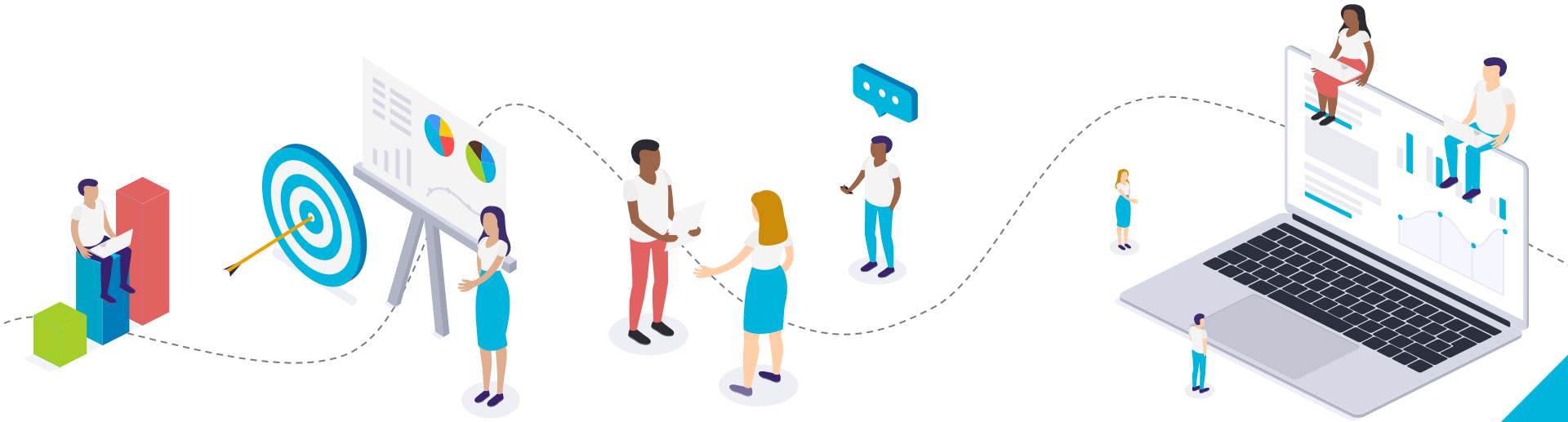
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Программа профессиональной переподготовки
«Технологии искусственного интеллекта, визуализации и анализа данных»



Задача регрессии – предсказание значений непрерывной целевой переменной

Линейная регрессионная модель: $\alpha(x) = \omega_0 + \sum_{j=1}^{d} x_j \omega_j = \omega_0 + \langle x, \omega \rangle$

Функционал ошибки: $Q(\alpha, X)$

Функция потерь: $L(\alpha, y)$

Измерение ошибок. Метрики качества

Зачем нужны метрики качества?

- Для задания функционала ошибки
- Для оценки качества итоговой модели
- Для подбора гиперпараметров

Измерение ошибок

Среднеквадратичное отклонение (MSE, mean squared error):

$$L(\alpha, y) = (\alpha(x_i) - y_i)^2$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\alpha(x_i) - y_i)^2$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\alpha(x_i) - y)^2}$$

- *MSE* подходит для контроля качества обучения (легко минимизировать)
- Плохо интерпретируется – сложно сделать выводы насколько хорошо модель решает задачу
- *MSE* сильно штрафует за большие ошибки (отклонения возводятся в квадрат), и алгоритм настраивается на выбросы

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_squared_error.html

Измерение ошибок

Коэффициент детерминации:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\alpha(x_i) - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Чем ближе значение коэффициента к 1, тем сильнее зависимость. При оценке регрессионных моделей это интерпретируется как соответствие модели данным. Модели с коэффициентом детерминации выше 80% можно признать достаточно хорошими. Равенство коэффициента детерминации единице означает, что объясняемая переменная в точности описывается рассматриваемой моделью.

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html

Измерение ошибок

Среднее абсолютное отклонение (MAE, mean absolute error):

$$L(\alpha, y) = |\alpha(x_i) - y_i|$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\alpha(x_i) - y_i|$$

- Сложнее минимизировать, чем MSE
- Менее чувствителен к выбросам, чем MSE

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_absolute_error.html

Измерение ошибок

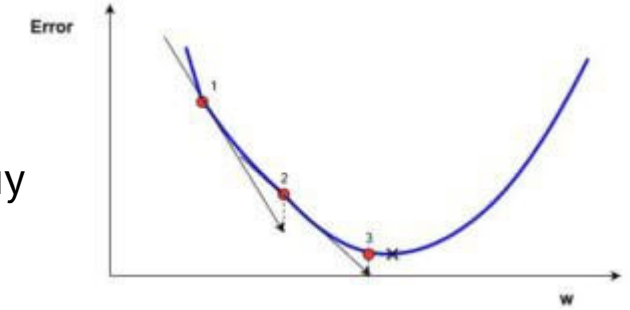
- MSLE (среднеквадратичная логарифмическая ошибка) – https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_squared_log_error.html#sklearn.metrics.mean_squared_log_error
- MAPE (средняя абсолютная процентная ошибка) – https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_absolute_percentage_error.html
- SMAPE (симметричная абсолютная процентная ошибка)

Ссылка на документацию: https://scikit-learn.org/stable/modules/model_evaluation.html#regression-metrics

Градиентный спуск

- *Градиентом* функции многих переменных называется вектор, координаты которого равны частным производным по соответствующим аргументам.
- *Градиент* – вектор частных производных Q по ω.

Градиент показывает направление наискорейшего роста функции, а антиградиент (градиент со знаком минус) в сторону *наискорейшего убывания*.



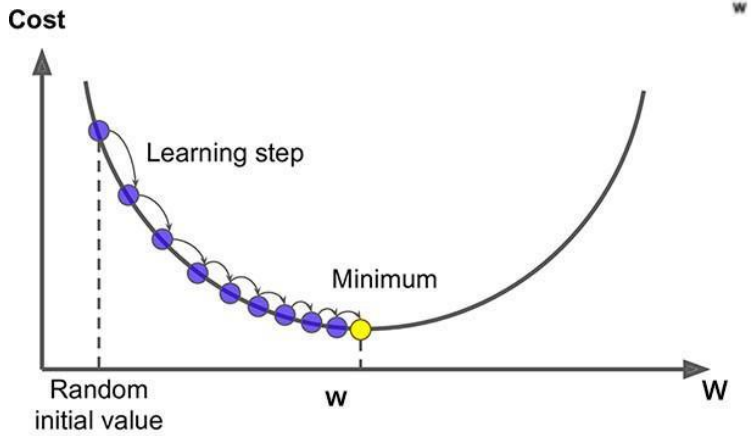
1. Задаем первое приближение для вектора весов ω (нули или маленькие случайные числа)
2. В цикле по t:

$$\omega^t = \omega^{t-1} - \eta_t \nabla Q(\omega^{t-1}, X)$$

η_t – шаг.

Если вектор весов меняется от шага к шагу не очень сильно, то наступила *сходимость*:

$$\|\omega^t - \omega^{t-1}\| < \varepsilon$$



Стохастический градиентный спуск

Проблема градиентного спуска заключается в том, что для того, чтобы определить новое приближение вектора весов необходимо вычислить градиент от каждого элемента выборки, что может сильно замедлять алгоритм. Идея ускорения алгоритма заключается в использовании только одного элемента для подсчета нового приближения весов:

$$\omega^t = \omega^{t-1} - \eta_t \nabla Q(\omega^{t-1}, \{x_i\})$$

Для больших массивов данных стохастический градиентный спуск может дать значительное преимущество в скорости по сравнению со стандартным градиентным спуском.

- https://scikit-learn.org/stable/modules/sgd.html
- Метод среднего стохастического градиента (SAG)

Оценка качества модели

Оценка качества модели не обучающих данных **не объективна**!

Как оценивать?

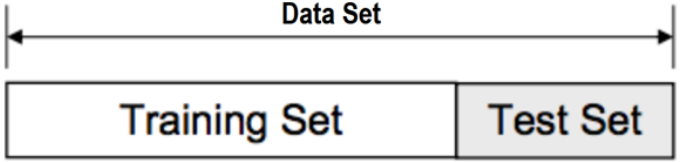
- С помощью тестовой (отложенной) выборки
- С помощью перекрестной проверки (кросс-валидация)

Оценка качества модели. Использование тестовой выборки

Выборку данных следует разбить выборку на две части:

- Первая будет использоваться для обучения
- Вторая для оценки качества

В каких пропорциях разделять? 70/30, 80/20



Результат сильно зависит от объектов, попавших в тестовую и обучающую выборку!

https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

```
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

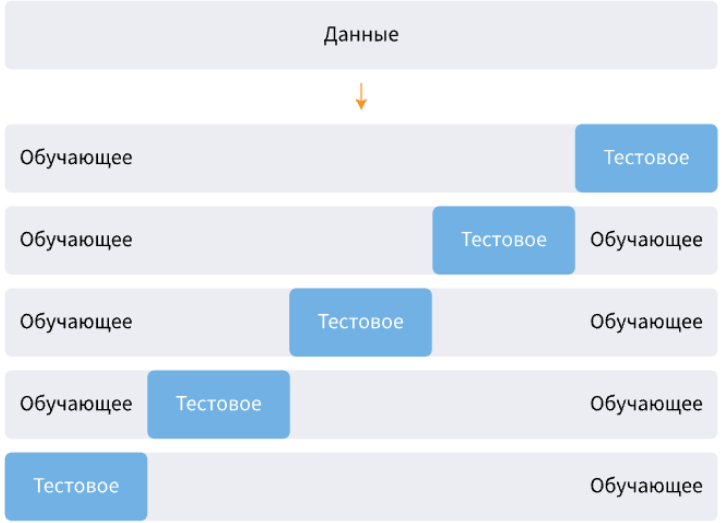
Размер тестовой выборки 30% от исходной

Если зафиксировать random_state, то при каждом запуске в выборку попадут одни и те же объекты

Перекрестная проверка

Метод заключается в разделении исходного множества данных на k примерно равных блоков. Затем на $k-1$ блоках производится обучение модели, а оставшийся блок используется для тестирования. Процедура повторяется k раз, при этом на каждом проходе для проверки выбирается новый блок, а обучение производится на оставшихся.

Нет конкретных рекомендаций относительно выбора k. Чем больше k, тем больше раз приходится обучать алгоритм. Поэтому на больших выборках следует выбирать небольшие значения k.



Перекрестная проверка

Перекрестная проверка имеет важное преимущество: если оценить ошибку модели на каждом блоке и усреднить ее по всем блокам, то полученная ее оценка будет более достоверной.

https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_val_score.html

```
from sklearn.model_selection import cross_val_score
cross_val_score(model, X, y, cv=3)
```

Количество блоков

Класс Kfold (https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.KFold.html) используется для того, чтобы разделить набор данных до моделирования, чтобы все модели использовали одни и те же разделения данных.