

Линейные методы регрессии

Паточенко Евгений
НИУ ВШЭ

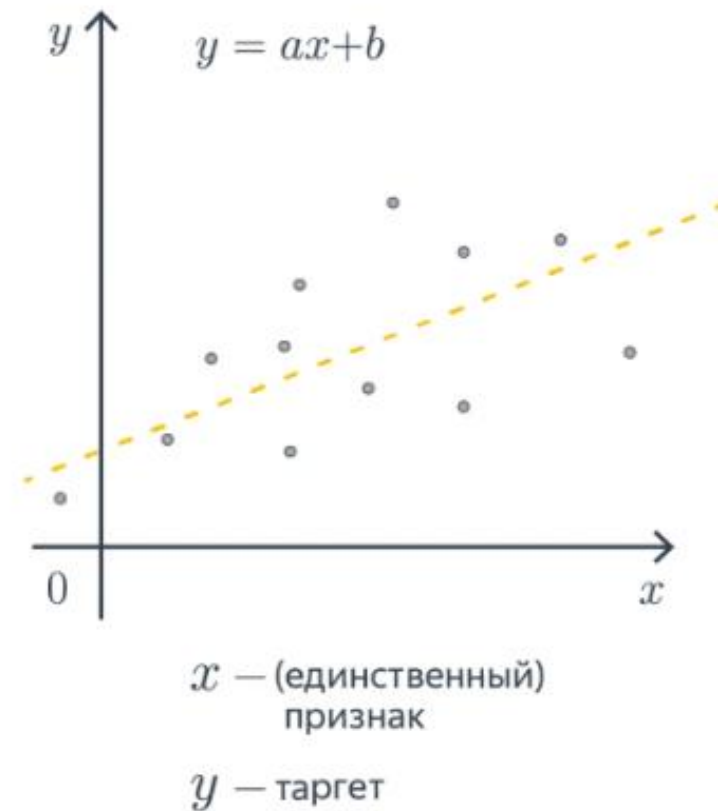
План занятия

- Линейная модель
- Оценка качества работы модели: функции потерь и метрики качества
- Переобучение

Линейная модель

Регрессия

Базовая модель машинного обучения и статистики, используемая для оценки зависимости между одной зависимой переменной (целевой) и одной или несколькими независимыми переменными (признаками), при этом целевая переменная — непрерывная величина



Линейная модель

Общий вид уравнения линейной регрессии

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n + \varepsilon,$$

где

- y — зависимая (целевая) переменная
- x_1, x_2, \dots, x_n — независимые переменные (признаки)
- β_0 — свободный член (intercept)
- $\beta_1, \beta_2, \dots, \beta_n$ — коэффициенты (веса) модели
- ε — ошибка модели (ошибка прогноза)

Линейная модель

Общий вид уравнения линейной регрессии

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n + \varepsilon,$$

где

- y — зависимая (целевая) переменная
- x_1, x_2, \dots, x_n — независимые переменные (признаки)
- β_0 — свободный член (intercept)
- $\beta_1, \beta_2, \dots, \beta_n$ — коэффициенты (веса) модели
- ε — ошибка модели (ошибка прогноза)

Сокращенная запись

$$y(x) = \beta_0 + \sum_{j=1}^n \beta_j x_j$$

Линейная модель

Общий вид уравнения линейной регрессии

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n + \varepsilon,$$

где

- y — зависимая (целевая) переменная
- x_1, x_2, \dots, x_n — независимые переменные (признаки)
- β_0 — свободный член (intercept)
- $\beta_1, \beta_2, \dots, \beta_n$ — коэффициенты (веса) модели
- ε — ошибка модели (ошибка прогноза)

Сокращенная запись

$$y(x) = \beta_0 + \sum_{j=1}^n \beta_j x_j$$

Запись через скалярное произведение

$$y(x) = \beta_0 \cdot 1 + \sum_{j=1}^n \beta_j x_j = \sum_{j=0}^n \beta_j x_j = (\beta, x)$$



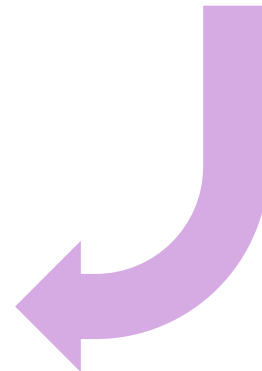
Линейная модель

Представление через скалярное произведение

Формула $y(x) = \beta_0 \cdot 1 + \sum_{j=1}^n \beta_j x_j = \sum_{j=0}^n \beta_j x_j = (\beta, x)$ — это не что иное, как скалярное произведение двух векторов:

- Вектора признаков объекта с фиктивной переменной $x = (1, x_1, \dots, x_n)$
- Вектора весов модели $\beta = (\beta_0, \dots, \beta_n)$

$$y(x, \beta) = \sum_{j=0}^n \beta_j x_j = (\beta, x)$$



Линейная модель

Пример простейшей линейной регрессии

Предположим, что в нашей компании работает очень ленивый маркетолог. Он решил простым, но модным способом защитить маркетинговый бюджет перед руководством, показав, что чем больше компания потратит на рекламу, тем большую выручку она получит.

Линейная модель

Пример простейшей линейной регрессии

Предположим, что в нашей компании работает очень ленивый маркетолог. Он решил простым, но модным способом защитить маркетинговый бюджет перед руководством, показав, что чем больше компания потратит на рекламу, тем большую выручку она получит.

Взял датасет с единственным признаком: бюджет на рекламу. Например, он будет выглядеть так



```
df = pd.read_csv('Датасет с затратами на рекламу.csv')
df
```

	ad_costs	revenue
0	8270	41794.49
1	1860	9548.03
2	6390	31977.79
3	6191	30985.12
4	6734	32950.27
...
495	5330	26892.59
496	9010	45122.65
497	7801	39861.79
498	5846	29735.07
499	7731	38556.87

500 rows x 2 columns

Линейная модель

Пример простейшей линейной регрессии

Сделал самую простую модель линейной регрессии

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
```

```
X = df[['ad_costs']]
y = df['revenue']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

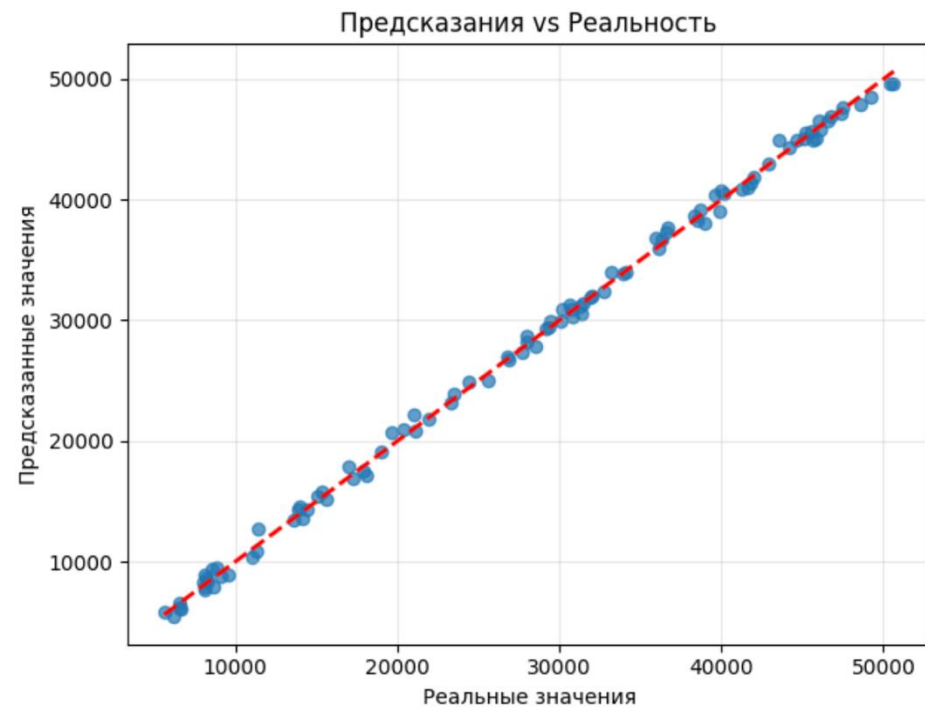
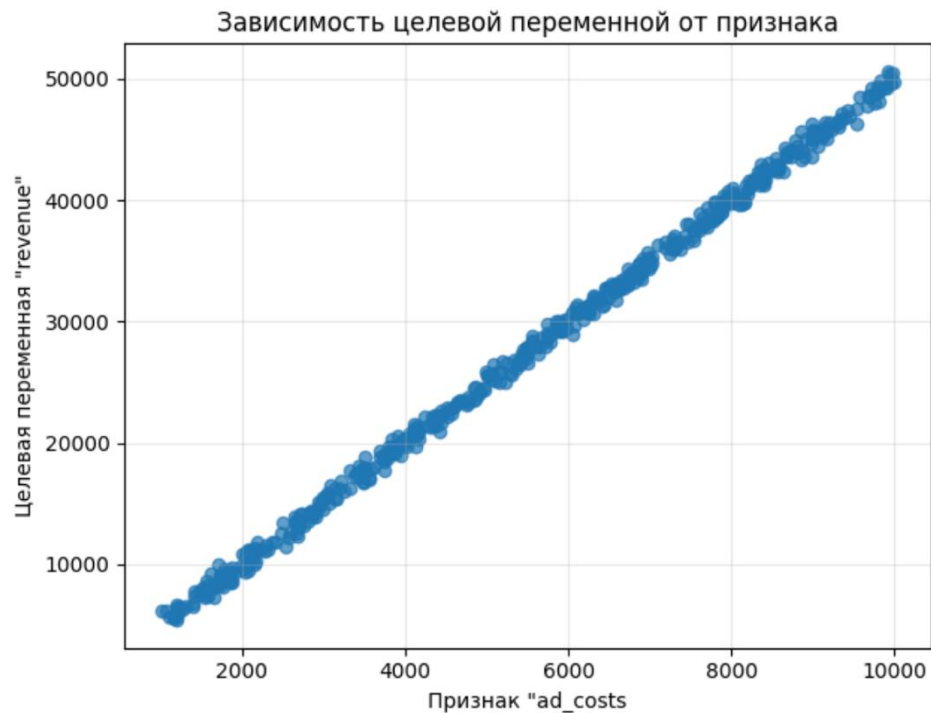
```
model = LinearRegression()
model.fit(X_train, y_train)
```

```
y_pred = model.predict(X_test)
```

Линейная модель

Пример простейшей линейной регрессии

Что значит линейной?




Линейная модель

Пример простейшей линейной регрессии

В конце наш маркетолог прогнал модель на трех вариантах бюджета и вывел красивые предсказания:

```
revenue = model.predict(budget_options)

result_table = pd.DataFrame({
    'budget_options': budget_options['ad_costs'].values,
    'revenue': revenue
})
result_table.round(1)
```



	budget_options	revenue
0	3000	15069.1
1	5000	25044.1
2	10000	49981.6

Линейная модель

В реальности так не бывает:)

Линейная модель

В реальности датасет будет скорее выглядеть так:

	ad_budget	marketing_spend	operational_costs	season	weekday	marketing_channel	product_category	day_of_month	month	holiday	promotion
0	16795	13546	18524	Summer	Monday	Social Media	Home	8	9	0	0
1	1860	3090	6882	Winter	Saturday	Influencer	Electronics	9	7	0	0
2	39158	17947	16113	Summer	Thursday	Search Ads	Food	13	8	0	0
3	45732	4824	15645	Winter	Wednesday	Influencer	Beauty	9	1	0	1
4	12284	11287	8012	Winter	Friday	Radio	Clothing	26	11	0	0
...
995	28139	9406	29272	Spring	Friday	Influencer	Books	8	5	1	1
996	12613	4646	17149	Summer	Friday	Email	Books	21	2	1	0
997	11589	12546	14150	Summer	Sunday	Radio	Home	24	2	0	0
998	14918	13111	9183	Winter	Tuesday	TV	Electronics	6	9	0	0
999	21119	13576	23632	Summer	Thursday	Email	Books	21	2	1	0

1000 rows x 19 columns

weekend	customer_rating	competitor_price	social_media_mentions	special_campaign	customer_feedback_score	delivery_time_days	revenue
1	3.0	158.757493	5.0	NaN	8.063535	NaN	17374.366924
0	3.0	151.788430	3.0	1.0	NaN	NaN	11918.449288
0	NaN	124.876416	6.0	NaN	4.429636	NaN	21210.075190
1	3.0	140.409655	6.0	NaN	7.971042	NaN	17485.981691
0	NaN	13.487588	4.0	NaN	4.756073	NaN	9667.649004
...
0	2.0	43.637874	NaN	NaN	4.580340	NaN	12873.641733
0	2.0	130.460146	2.0	NaN	NaN	NaN	15284.183813
0	2.0	79.189036	5.0	NaN	3.977666	NaN	14615.520120
1	4.0	171.949416	6.0	NaN	NaN	11.0	18678.260756
0	3.0	153.745005	7.0	NaN	NaN	10.0	13470.416949

Линейная модель

Что мы можем сказать о данных на этом этапе?

```
df.describe().T
```

	count	mean	std	min	25%	50%	75%	max
ad_budget	1000.0	24923.625000	14345.114665	1009.000000	12106.500000	24536.000000	38223.250000	49984.000000
marketing_spend	1000.0	10105.865000	5650.980217	509.000000	5136.000000	9985.500000	14799.250000	19965.000000
operational_costs	1000.0	16199.290000	8133.536465	2016.000000	9095.000000	16113.500000	23595.500000	29999.000000
day_of_month	1000.0	15.714000	8.581483	1.000000	8.000000	16.000000	23.000000	30.000000
month	1000.0	6.489000	3.416658	1.000000	4.000000	6.000000	10.000000	12.000000
holiday	1000.0	0.204000	0.403171	0.000000	0.000000	0.000000	0.000000	1.000000
promotion	1000.0	0.274000	0.446232	0.000000	0.000000	0.000000	1.000000	1.000000
weekend	1000.0	0.271000	0.444699	0.000000	0.000000	0.000000	1.000000	1.000000
customer_rating	859.0	2.968568	1.179739	1.000000	2.000000	3.000000	4.000000	5.000000
competitor_price	912.0	107.361706	54.953716	10.045766	59.589899	107.465442	155.014633	199.905986
social_media_mentions	903.0	4.941307	2.242477	0.000000	3.000000	5.000000	6.000000	15.000000
special_campaign	76.0	0.407895	0.494709	0.000000	0.000000	0.000000	1.000000	1.000000
customer_feedback_score	791.0	5.577914	2.555678	1.081848	3.385418	5.557296	7.746434	9.990487
delivery_time_days	99.0	7.212121	3.785776	1.000000	4.000000	7.000000	11.000000	13.000000
revenue	1000.0	15189.590056	5571.866469	1348.970818	11060.541262	14768.856798	18734.404946	33076.805718

```
df.describe(include=object).T
```

	count	unique	top	freq
season	1000	4	Winter	263
weekday	1000	7	Monday	161
marketing_channel	928	6	Search Ads	174
product_category	940	6	Food	169

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 19 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   ad_budget                            1000 non-null   int64
1   marketing_spend                      1000 non-null   int64
2   operational_costs                    1000 non-null   int64
3   season                              1000 non-null   object
4   weekday                             1000 non-null   object
5   marketing_channel                    928 non-null    object
6   product_category                     940 non-null    object
7   day_of_month                         1000 non-null   int64
8   month                               1000 non-null   int64
9   holiday                             1000 non-null   int64
10  promotion                           1000 non-null   int64
11  weekend                              1000 non-null   int64
12  customer_rating                      859 non-null    float64
13  competitor_price                     912 non-null    float64
14  social_media_mentions                 903 non-null    float64
15  special_campaign                      76 non-null     float64
16  customer_feedback_score               791 non-null    float64
17  delivery_time_days                    99 non-null     float64
18  revenue                              1000 non-null   float64
dtypes: float64(7), int64(8), object(4)
memory usage: 148.6+ KB
```

Линейная модель

Что нам нужно сделать прежде чем обучить модель?

Линейная модель

Что нам нужно сделать прежде чем обучить модель?

- Обработать пропуски (какие стратегии вы помните?)

Линейная модель

Что нам нужно сделать прежде чем обучить модель?

- Обработать пропуски (какие стратегии вы помните?)
- Закодировать категориальные признаки

Линейная модель

Что нам нужно сделать прежде чем обучить модель?

- Обработать пропуски (какие стратегии вы помните?)
- Закодировать категориальные признаки
- Посмотреть распределения и зависимости, матрицу корреляций (зачем?)

Линейная модель

Что нам нужно сделать прежде чем обучить модель?

- Обработать пропуски (какие стратегии вы помните?)
- Закодировать категориальные признаки
- Посмотреть распределения и зависимости, матрицу корреляций (зачем?)
- После этого, возможно, еще обработать данные (что еще может быть нужно?)

Линейная модель

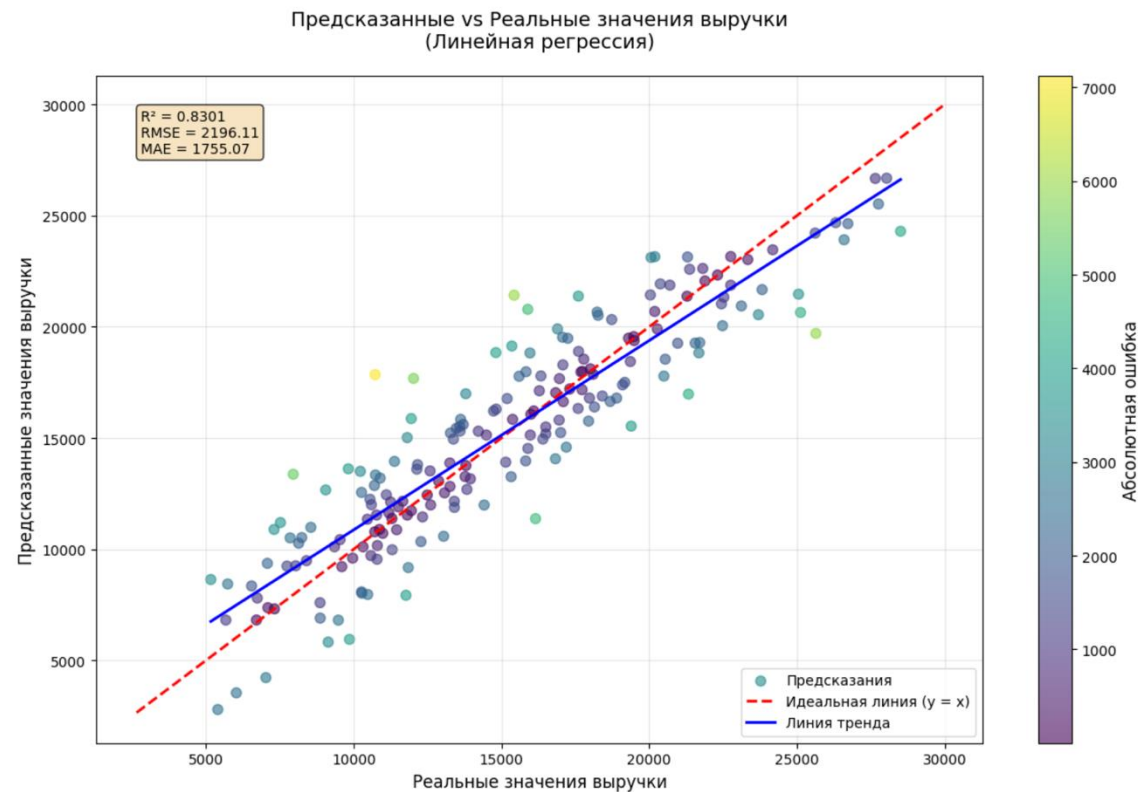
Что нам нужно сделать прежде чем обучить модель?

- Обработать пропуски (какие стратегии вы помните?)
- Закодировать категориальные признаки
- Посмотреть распределения и зависимости, матрицу корреляций (зачем?)
- После этого, возможно, еще обработать данные (что еще может быть нужно?)

Подробно посмотрим все эти действия на практике во второй части занятия!

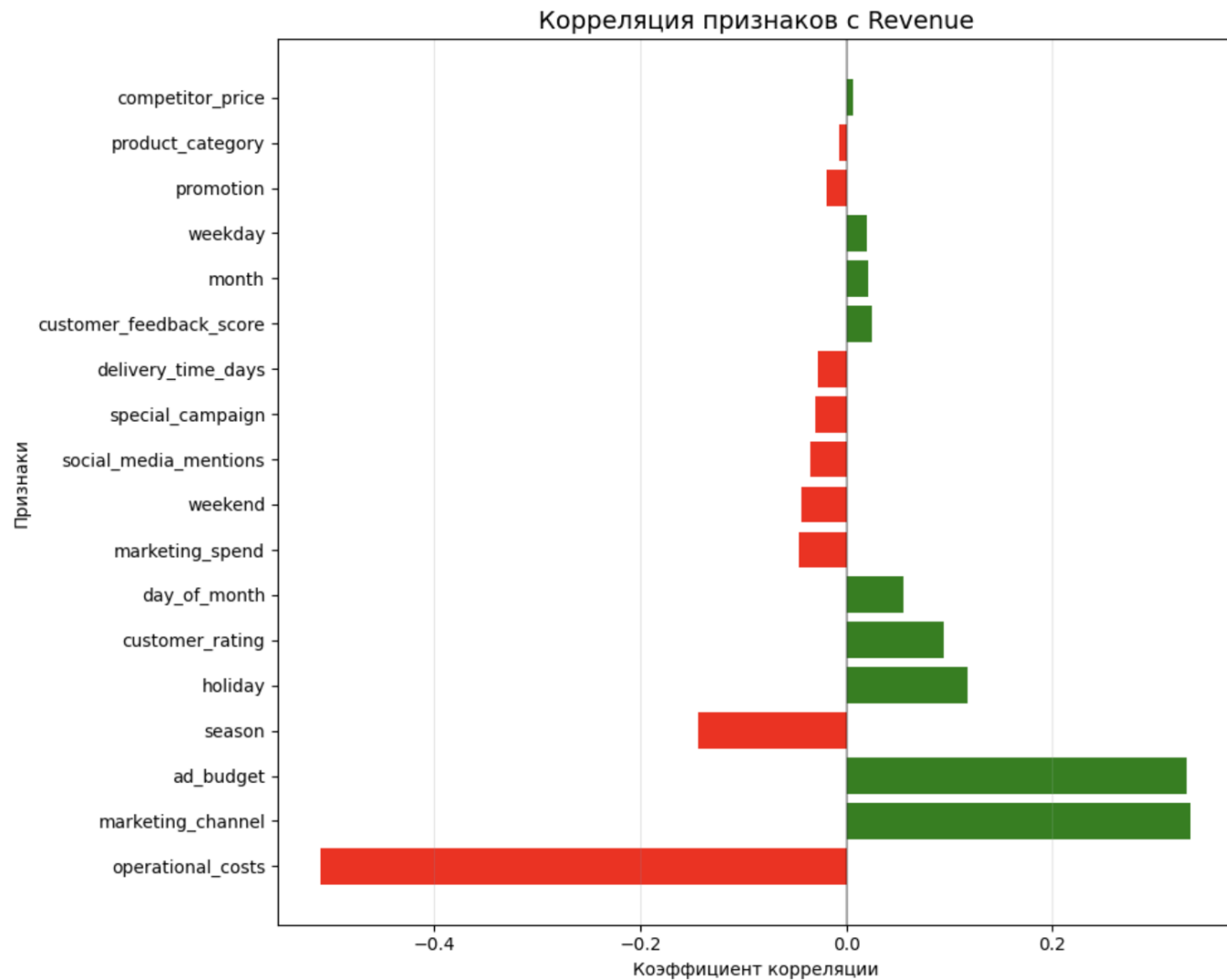
Линейная модель

Только после тщательной обработки данных обучаем модель



Линейная модель

Заодно оказалось, что расходы на рекламу в нашей компании вносят не такой сильный вклад в размер выручки, как того хотел бы ленивый маркетолог:)



А как понять, что нашей модели можно доверять?

Оценка качества модели

Функции потерь и метрики качества (напоминание)

Функция потерь — функция, измеряющая качество работы алгоритма (ошибку предсказания модели) на объектах обучающей выборки.

Пример: MSE (среднеквадратичная ошибка, разница между предсказанным значением и истинным, возведенная в квадрат)

Метрика качества — функция, измеряющая качество работы алгоритма (ошибку предсказания модели) на данных, которые модель еще не видела, и используемая для сравнения моделей.

Примеры: MAE (абсолютная ошибка, модуль разницы между предсказанным значением и истинным), RMSE (корень из MSE)

Оценка качества модели

Линейная регрессия:

$$a(x) = \beta_0 + \sum_{i=1}^n \beta_i x_i = (\beta, x)$$

Обучение линейной регрессии — это минимизация MSE

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n (a(x_i) - y_i)^2 = \frac{1}{n} \sum_{i=1}^n ((\beta, x_i) - y_i)^2 \rightarrow \min_{\beta}$$

Оценка качества модели

Метод наименьших квадратов (OLS)

Цель — минимизировать сумму квадратов ошибок: $\min_{\beta} \|y - X\beta\|^2$

Решение имеет аналитическую форму: $\hat{\beta} = (X^T X)^{-1} X^T y$

Оценка качества модели

Метод наименьших квадратов (OLS)

Цель — минимизировать сумму квадратов ошибок: $\min_{\beta} \|y - X\beta\|^2$

Решение имеет аналитическую форму: $\hat{\beta} = (X^T X)^{-1} X^T y$

Недостатки аналитической формулы:

Оценка качества модели

Метод наименьших квадратов (OLS)

Цель — минимизировать сумму квадратов ошибок: $\min_{\beta} \|y - X\beta\|^2$

Решение имеет аналитическую форму: $\hat{\beta} = (X^T X)^{-1} X^T y$

Недостатки аналитической формулы:

- Обращение матрицы — сложная операция ($O(N^3)$ от числа признаков)
- Матрица $X^T X$ может быть вырожденной или плохо обусловленной
- Если заменить среднеквадратичный функционал ошибки на другой, то скорее всего не найдем аналитическое решение

Оценка качества модели

Почему MSE? Вероятностная постановка

Даже если целевая переменная линейно зависит от признаков, идеальной модели (с вероятностью 1) не существует, то есть реальные ответы будут несильно, но отличаться от предсказаний. Поэтому мы пишем

$$y \approx (\beta, x)$$

Второй подход заключается в том, что мы объясняем неидеальность прогноза неполнотой данных, или же шумами в данных. Тогда мы пишем так:

$$y = (\beta, x) + \varepsilon,$$

где ε — это шум в данных

Оценка качества модели

Почему MSE? Вероятностная постановка

Шум в данных обычно имеет некоторое распределение. В большинстве реальных задач считается, что

$$\varepsilon \sim N(0, \sigma^2)$$

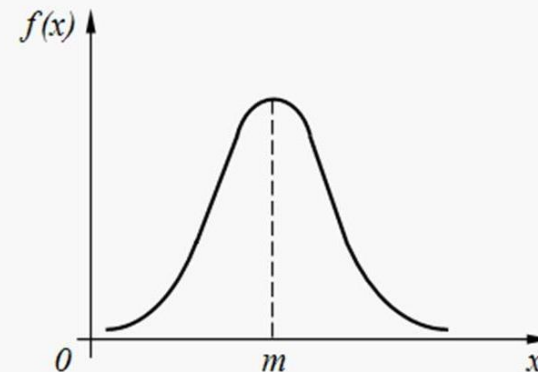
Отсюда получаем, что

$$y \sim N((\beta, x), \sigma^2)$$

Это означает, что вероятность наблюдать y при данных значениях x равна

$$p(y|x, \beta) \sim N((\beta, x), \sigma^2)$$

График плотности нормального распределения



Оценка качества модели

Метод максимального правдоподобия

Мы хотим подобрать такой вектор β , что вероятность наблюдать некоторое значение y при наблюдаемых x максимальна

$$p(y|X, \beta) \rightarrow \max_{\beta}$$

Величина $p(y|X, \beta)$ называется функцией правдоподобия (или правдоподобием) выборки

Оценка качества модели

Метод максимального правдоподобия

Модель данных с некоррелированным гауссовским шумом:

$$y_i = (\beta, x_i) + \varepsilon_i, \varepsilon_i \sim N(0, \sigma^2), i = 1, \dots, l$$

Оценка качества модели

Метод максимального правдоподобия

Модель данных с некоррелированным гауссовским шумом:

$$y_i = (\beta, x_i) + \varepsilon_i, \varepsilon_i \sim N(0, \sigma^2), i = 1, \dots, l$$

Тогда

$$y_i \sim N((\beta, x_i), \sigma^2), i = 1, \dots, l$$

Оценка качества модели

Метод максимального правдоподобия

Модель данных с некоррелированным гауссовским шумом:

$$y_i = (\beta, x_i) + \varepsilon_i, \varepsilon_i \sim N(0, \sigma^2), i = 1, \dots, l$$

Тогда

$$y_i \sim N((\beta, x_i), \sigma^2), i = 1, \dots, l$$

Метод максимального правдоподобия (ММП):

$$L(y_1 \dots y_l | \beta) = \prod_{i=1}^l \frac{1}{\sigma \sqrt{2\pi}} \exp \left(-\frac{1}{2\sigma^2} (y_i - (\beta, x_i))^2 \right) \rightarrow \max_{\beta}$$

Оценка качества модели

Метод максимального правдоподобия

Модель данных с некоррелированным гауссовским шумом:

$$y_i = (\beta, x_i) + \varepsilon_i, \varepsilon_i \sim N(0, \sigma^2), i = 1, \dots, l$$

Тогда

$$y_i \sim N((\beta, x_i), \sigma^2), i = 1, \dots, l$$

Метод максимального правдоподобия (ММП):

$$L(y_1 \dots y_l | \beta) = \prod_{i=1}^l \frac{1}{\sigma \sqrt{2\pi}} \exp \left(-\frac{1}{2\sigma^2} (y_i - (\beta, x_i))^2 \right) \rightarrow \max_{\beta}$$

$$-\ln L(y_1 \dots y_l | \beta) = cnst + \frac{1}{2\sigma^2} \sum_{i=1}^l (y_i - (\beta, x_i))^2 \rightarrow \min_{\beta}$$

Оценка качества модели

Метод максимального правдоподобия

Модель данных с некоррелированным гауссовским шумом:

$$y_i = (\beta, x_i) + \varepsilon_i, \varepsilon_i \sim N(0, \sigma^2), i = 1, \dots, l$$

Тогда

$$y_i \sim N((\beta, x_i), \sigma^2), i = 1, \dots, l$$

Метод максимального правдоподобия (ММП):

$$L(y_1 \dots y_l | \beta) = \prod_{i=1}^l \frac{1}{\sigma \sqrt{2\pi}} \exp \left(-\frac{1}{2\sigma^2} (y_i - (\beta, x_i))^2 \right) \rightarrow \max_{\beta}$$

$$-\ln L(y_1 \dots y_l | \beta) = \text{cnst} + \frac{1}{2\sigma^2} \sum_{i=1}^l (y_i - (\beta, x_i))^2 \rightarrow \min_{\beta}$$

В данном случае ММП
совпадает с МНК

Оценка качества модели

Еще раз про функции потерь и метрики качества

- Функция потерь (функционал ошибки) — функция, которую минимизируют в процессе обучения модели для нахождения неизвестных параметров (весов)
- Метрика качества — функция, которую используют для оценки качества обученной модели.

Оценка качества модели

Еще раз про функции потерь и метрики качества

- Функция потерь (функционал ошибки) — функция, которую минимизируют в процессе обучения модели для нахождения неизвестных параметров (весов)
- Метрика качества — функция, которую используют для оценки качества обученной модели.

Одна и та же функция может выступать и функцией потерь, и метрикой качества — выбор зависит от задачи.

Оценка качества модели

Метрики качества

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

Насколько метрика MSE интерпретируема?

Оценка качества модели

Метрики качества

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

Насколько метрика MSE интерпретируема?

MSE подходит для сравнения двух моделей или для контроля качества во время обучения, но не позволяет сделать выводы о том, насколько хорошо модель решает задачу

$MSE = 100 \text{ кг}^2$ — это как?:)

Оценка качества модели

Метрики качества

RMSE (root mean squared error) — корень из MSE

$$RMSE(a, X) = \sqrt{\frac{1}{n} \sum_{i=1}^n (a(x_i)) - y_i)^2}$$

- ⊕ Решает проблему единиц измерения
- ⊖ Все еще тяжело понять, насколько хорошо модель решает задачу, так как метрика не ограничена сверху

Оценка качества модели

Метрики качества

Коэффициент детерминации R^2 — нормализованная MSE

$$R^2(a, X) = 1 - \frac{MSE(a, Y)}{D[Y]}, \text{ где } D[Y] \text{ — дисперсия целевой переменной}$$

- ⊕ Характеризует долю дисперсии целевой переменной, объясняемую моделью
 - Чем ближе R^2 к 1, тем лучше модель объясняет данные
 - Чем ближе R^2 к 0, тем ближе модель к константному решению
 - Отрицательный R^2 — модель хуже константного решения и плохо объясняет данные
- ⊖ Может быть непонятна для бизнес-заказчика

Оценка качества модели

Метрики качества

MAE (mean absolute error) — средняя абсолютная ошибка

$$MAE(a, X) = \sum_{i=0}^n |a(x_i) - y_i|$$

- ⊕ В отличие от MSE, которая из-за возведения в квадрат делает особый акцент на объектах с большей ошибкой, MAE менее чувствительна к выбросам
- ⊖ Функция модуля не дифференцируема, сложно производить оптимизацию напрямую

Оценка качества модели

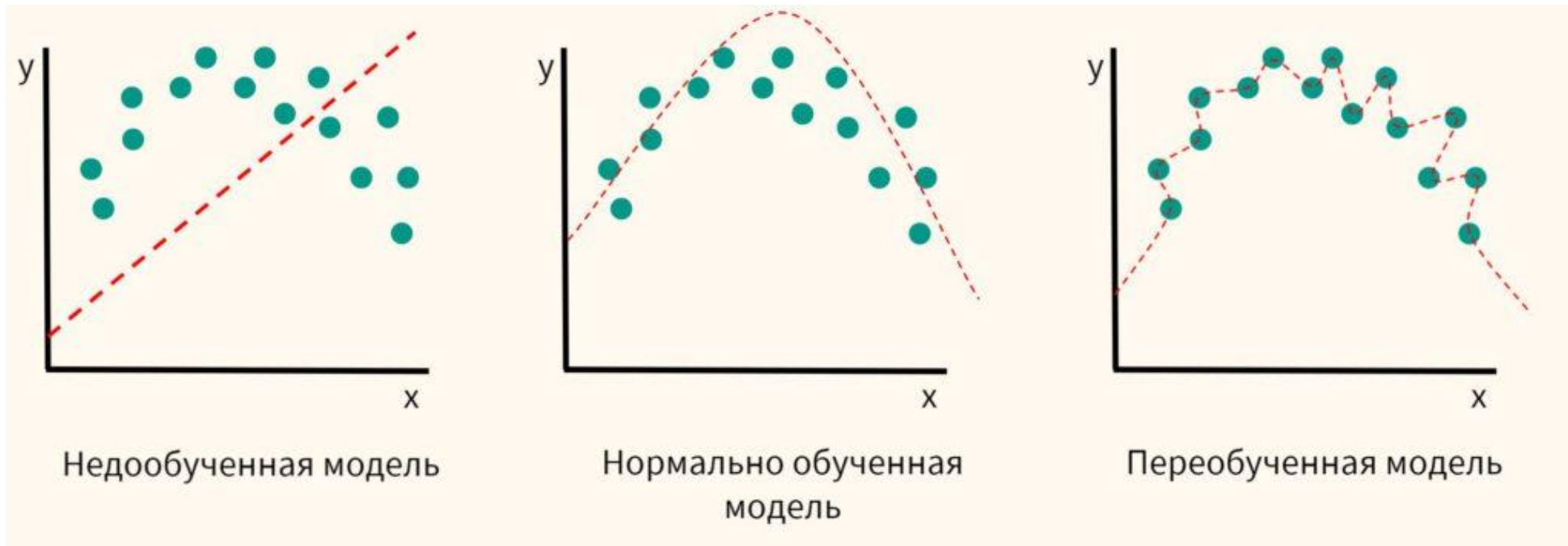
d2_absolute_error_score	D^2 regression score function, fraction of absolute error explained.
d2_pinball_score	D^2 regression score function, fraction of pinball loss explained.
d2_tweedie_score	D^2 regression score function, fraction of Tweedie deviance explained.
explained_variance_score	Explained variance regression score function.
max_error	The max_error metric calculates the maximum residual error.
mean_absolute_error	Mean absolute error regression loss.
mean_absolute_percentage_error	Mean absolute percentage error (MAPE) regression loss.
mean_gamma_deviance	Mean Gamma deviance regression loss.
mean_pinball_loss	Pinball loss for quantile regression.
mean_poisson_deviance	Mean Poisson deviance regression loss.
mean_squared_error	Mean squared error regression loss.
mean_squared_log_error	Mean squared logarithmic error regression loss.
mean_tweedie_deviance	Mean Tweedie deviance regression loss.
median_absolute_error	Median absolute error regression loss.
r2_score	R^2 (coefficient of determination) regression score function.
root_mean_squared_error	Root mean squared error regression loss.
root_mean_squared_log_error	Root mean squared logarithmic error regression loss.

... и это только для линейной регрессии

Метрики реализованы в библиотеке [scikit-learn](#)

Переобучение

Переобучение (*overfitting*) — явление, при котором качество модели на новых данных сильно хуже, чем на обучающей выборке.



Переобучение

Причины переобучения

Переобучение

Причины переобучения

- Переобучение есть всегда, когда есть оптимизация параметров по конечной (заведомо неполной) выборке

Переобучение

Причины переобучения

- Переобучение есть всегда, когда есть оптимизация параметров по конечной (заведомо неполной) выборке
- При небольшом количестве объектов в датасете выше вероятность того, что модель запомнит обучающие данные, а не будет обобщать их на новые, ранее невиданные экземпляры

Переобучение

Причины переобучения

- Переобучение есть всегда, когда есть оптимизация параметров по конечной (заведомо неполной) выборке
- При небольшом количестве объектов в датасете выше вероятность того, что модель запомнит обучающие данные, а не будет обобщать их на новые, ранее невиданные экземпляры
- Избыточная сложность модели (большое количество весов). В этом случае лишние степени свободы в модели «тратятся» на чрезмерно точную подгонку под обучающую выборку

Переобучение

Причины переобучения

- Переобучение есть всегда, когда есть оптимизация параметров по конечной (заведомо неполной) выборке
- При небольшом количестве объектов в датасете выше вероятность того, что модель запомнит обучающие данные, а не будет обобщать их на новые, ранее невиданные экземпляры
- Избыточная сложность модели (большое количество весов). В этом случае лишние степени свободы в модели «тратятся» на чрезмерно точную подгонку под обучающую выборку

С переобучением можно и нужно бороться! Как это сделать — рассмотрим на следующих занятиях.

Подведение итогов

- Линейная регрессия — базовый метод машинного обучения для моделирования зависимости между признаками и целевой переменной
- Ключевой этап — предобработка признаков. Критически важен для итогового качества модели
- Важно верно подобрать метрику качества, смотреть несколько метрик в зависимости от задачи
- Важная проблема — переобучение: модель может запоминать данные вместо обобщения, особенно на малом датасете или при избыточной сложности