

Машинное обучение

Часть I

Махоткин Даниил Русланович



2020

Содержание курса

Лекция 1

Intro to ML, Как все началось,
Постановка задачи, простейшие
методы, валидация моделей, kNN

Лекция 2

Линейная регрессия, градиентный
спуск, регуляризация,
классификация, мультикласс

Лекция 3-

Deep Learning
Нейронные сети
Обучение нейронных сетей
CNN: Сверточные нейронные сети.

Давайте решим задачу

Мальчик на санках едет с горки. Масса мальчика вместе с санками составляет 40 кг, угол наклона горы 30° . Найдите ускорение, которым съезжает мальчик, если коэффициент трения скольжения равен 0,2.

Давайте решим задачу

Мальчик на санках едет с горки. Масса мальчика вместе с санками составляет 40 кг, угол наклона горы 30° . Найдите ускорение, с которым съезжает мальчик, если коэффициент трения скольжения равен 0,2.

Дано:

$$m = 40 \text{ кг}$$

$$\alpha = 30^\circ$$

$$\mu = 0,2$$

$$a - ?$$

$$m\vec{a} = \vec{N} + m\vec{g} + \vec{F}_{\text{тр}}$$

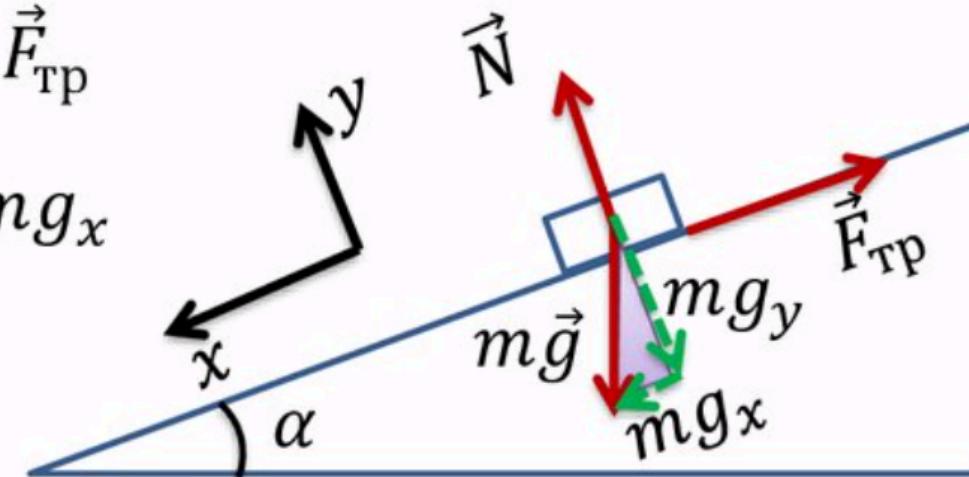
$$X: ma = -F_{\text{тр}} + mg_x$$

$$Y: 0 = N - mg_y$$

$$N = mg_y$$

$$F_{\text{тр}} = \mu N = \mu mg_y$$

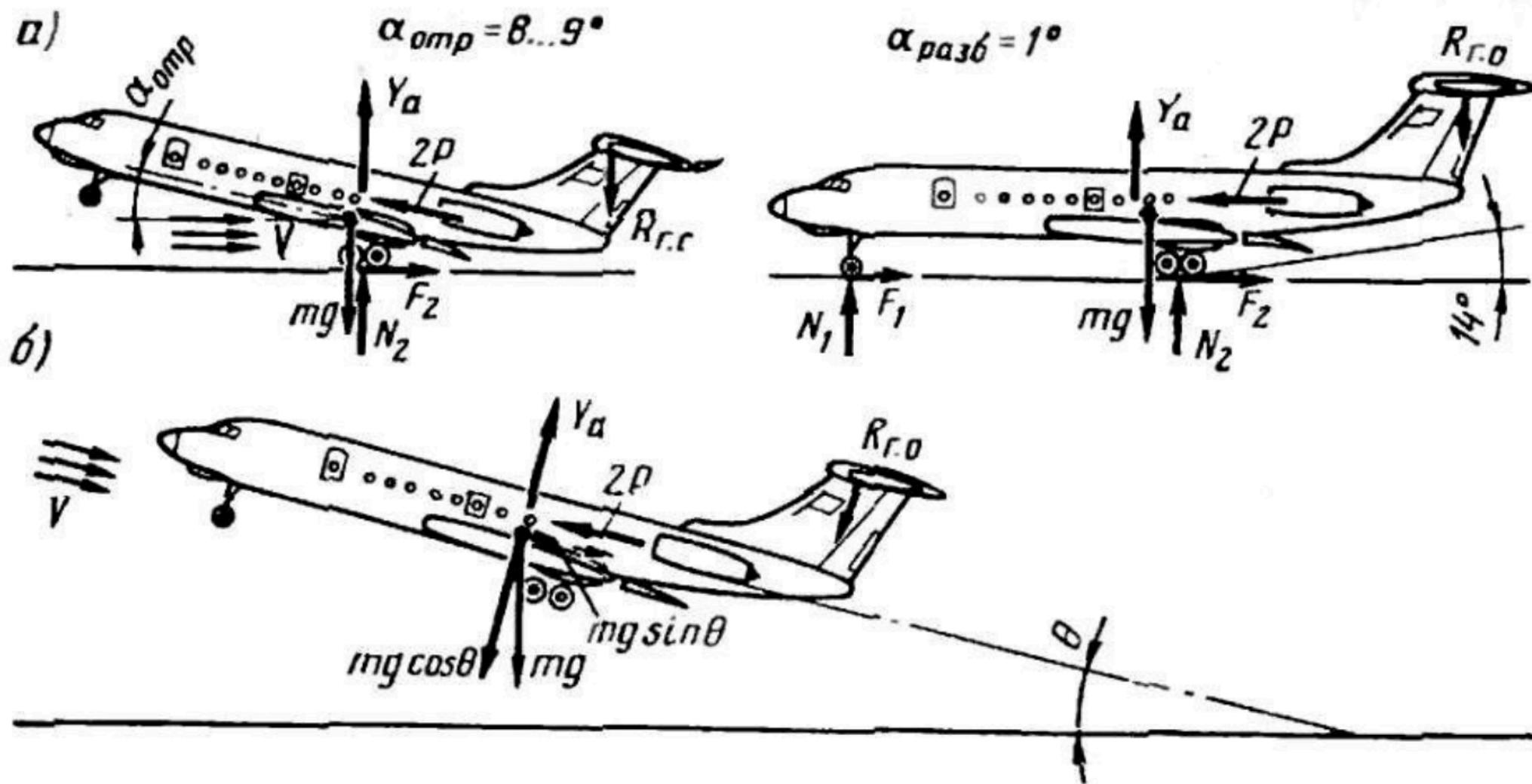
$$ma = mg_x - \mu mg_y$$



$$mg_x = mg \sin \alpha$$

$$mg_y = mg \cos \alpha$$

А если система сложнее?



Давайте решим задачу

Дано:

Сложная система

1000 или даже 5000 параметров

Мы не имеем представления, как эти параметры влияют на наше решение?



Можем ли мы сделать так?

проводить миллион экспериментов

показать результаты компьютеру

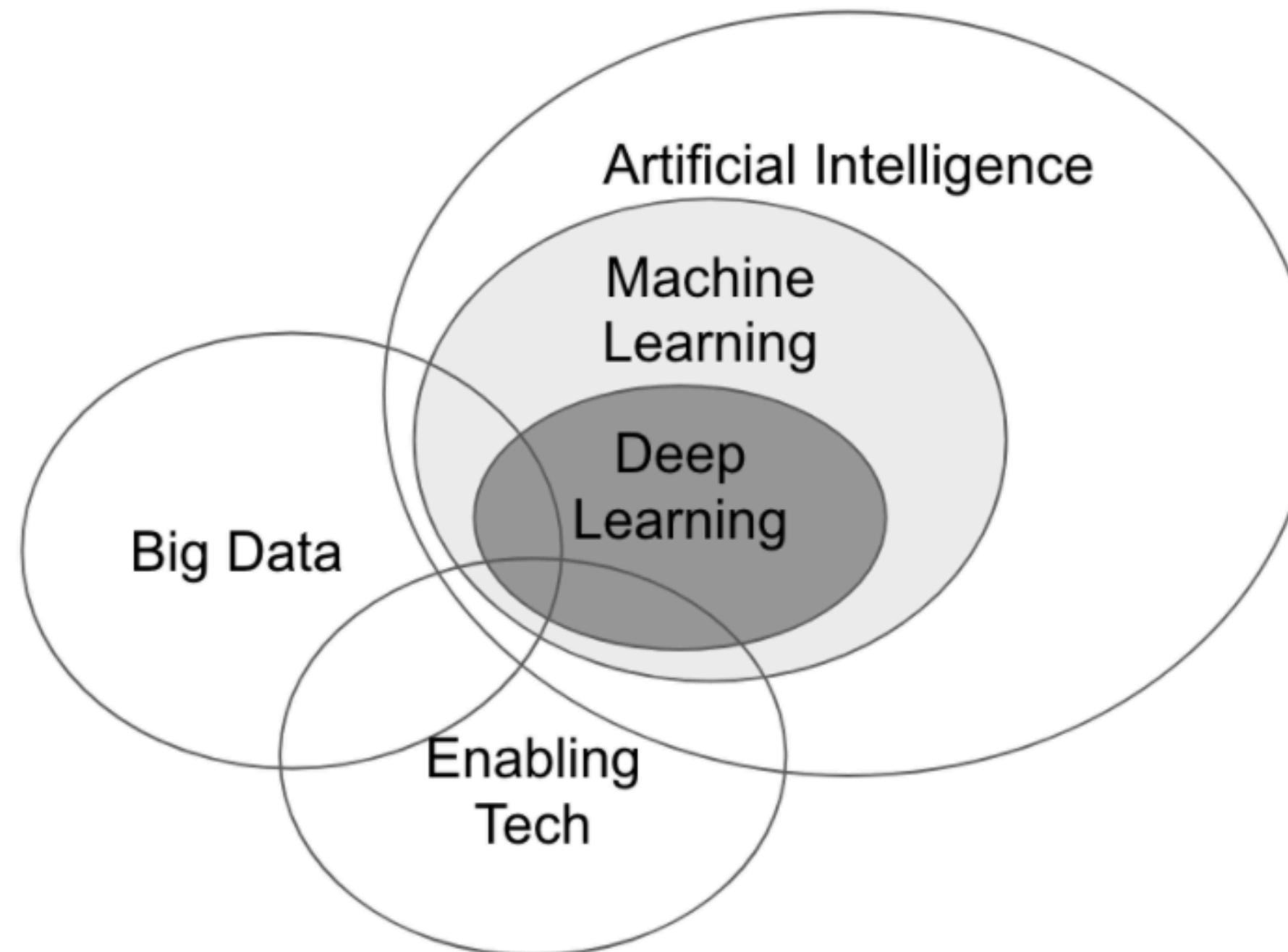
Компьютер сам найдет все связи и зависимости
а мы Будем получать ответы на любых новых данных

Где взять данные?

- [Google Dataset Search](#). Dataset Search позволяет по ключевому слову искать датасеты по всей Сети.
- [Kaggle](#). Площадка для соревнований по машинному обучению с множеством интересных датасетов. В [списке датасетов](#) можно найти разные нишевые экземпляры — от [оценок рамена](#) до [баскетбольных данных NCAA](#) и [базы лицензий на домашних животных в Сиэтле](#).
- [UCI Machine Learning Repository](#). Один из старейших источников датасетов в Сети и первое место, куда стоит заглянуть в поиске интересных датасетов. Хотя они добавляются пользователями и потому имеют различную степень «чистоты», большинство из них очищены. Данные можно скачивать сразу, без регистрации.
- [VisualData](#). Датасеты для компьютерного зрения, разбитые по категориям. Доступен поиск.
- [Find Datasets | CMU Libraries](#). Коллекция датасетов, предоставленная университетом Карнеги Меллон.

Больше датасетов: <https://tproger.ru/translations/the-best-datasets-for-machine-learning-and-data-science/>

Место машинного обучения в области ИИ



Историческая справка

ГОД

СОБЫТИЕ

1763	Опубликовано эссе Томаса Байеса, представляющее работу, лежащую в основе Теоремы Байеса.
1805	Лежандр описывает метод наименьших квадратов.
1812	Лаплас публикует работу, в которой определена Теорема Байеса.
1913	Андрей Марков описывает метод, позже называемый «Цепи Маркова».
1950	Алан Тьюринг предлагает концепцию Машинного обучения, предвмещающую генетические алгоритмы.
1957	Розенблат изобретает perceptron.
1967	Изобретен метод ближайших соседей
1970	Seppo Linnainmaa публикует общий метод автоматического дифференцирования (AD)
1986	Seppo Linnainmaa применяе обратный режим автоматического дифференцирования

Теорема Байеса

Формула Байеса:

$$P(A | B) = \frac{P(B | A) P(A)}{P(B)},$$

где

$P(A)$ – априорная вероятность гипотезы A (смысл такой терминологии см. ниже);

$P(A | B)$ – вероятность гипотезы A при наступлении события B (апостериорная вероятность);

$P(B | A)$ – вероятность наступления события B при истинности гипотезы A ;

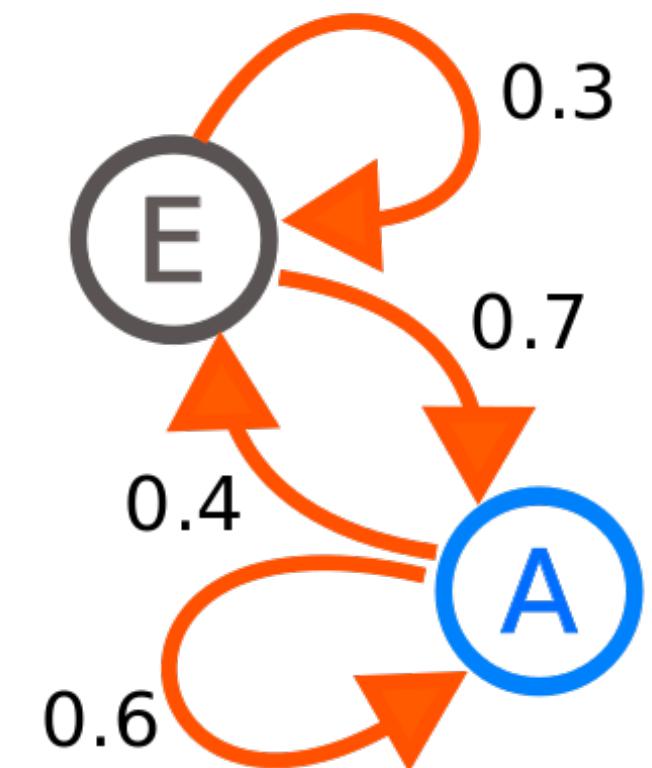
$P(B)$ – полная вероятность наступления события B .

Цепь Маркова

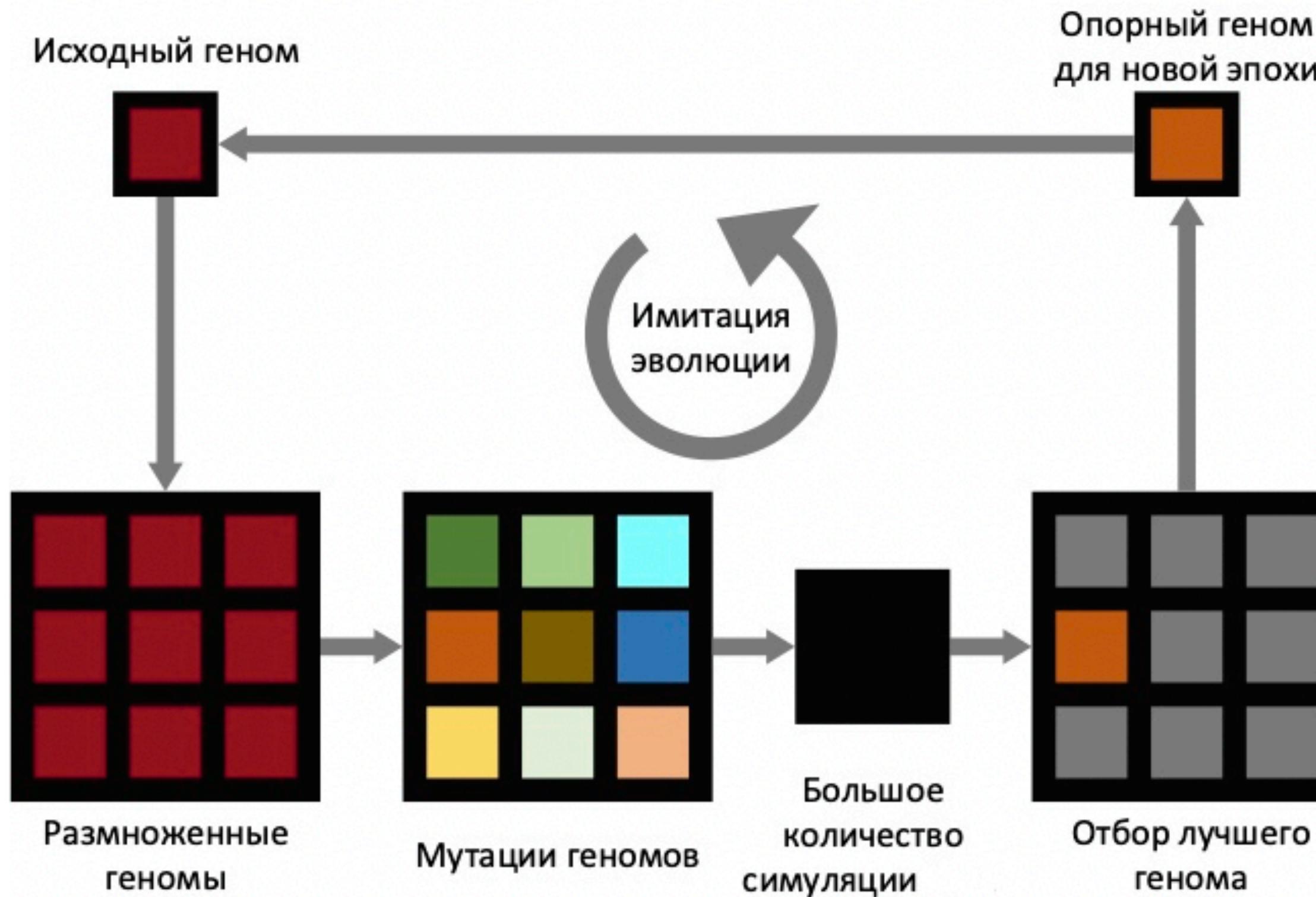
Цéпь Мárкова – последовательность случайных событий с конечным или счётым числом исходов, характеризующаяся тем свойством, что, говоря нестрого, при фиксированном настоящем будущее независимо от прошлого.

Последовательность дискретных случайных величин называется простой цепью Маркова (с дискретным временем), если:

$$\mathbb{P}(X_{n+1} = i_{n+1} \mid X_n = i_n, X_{n-1} = i_{n-1}, \dots, X_0 = i_0) = \mathbb{P}(X_{n+1} = i_{n+1} \mid X_n = i_n).$$

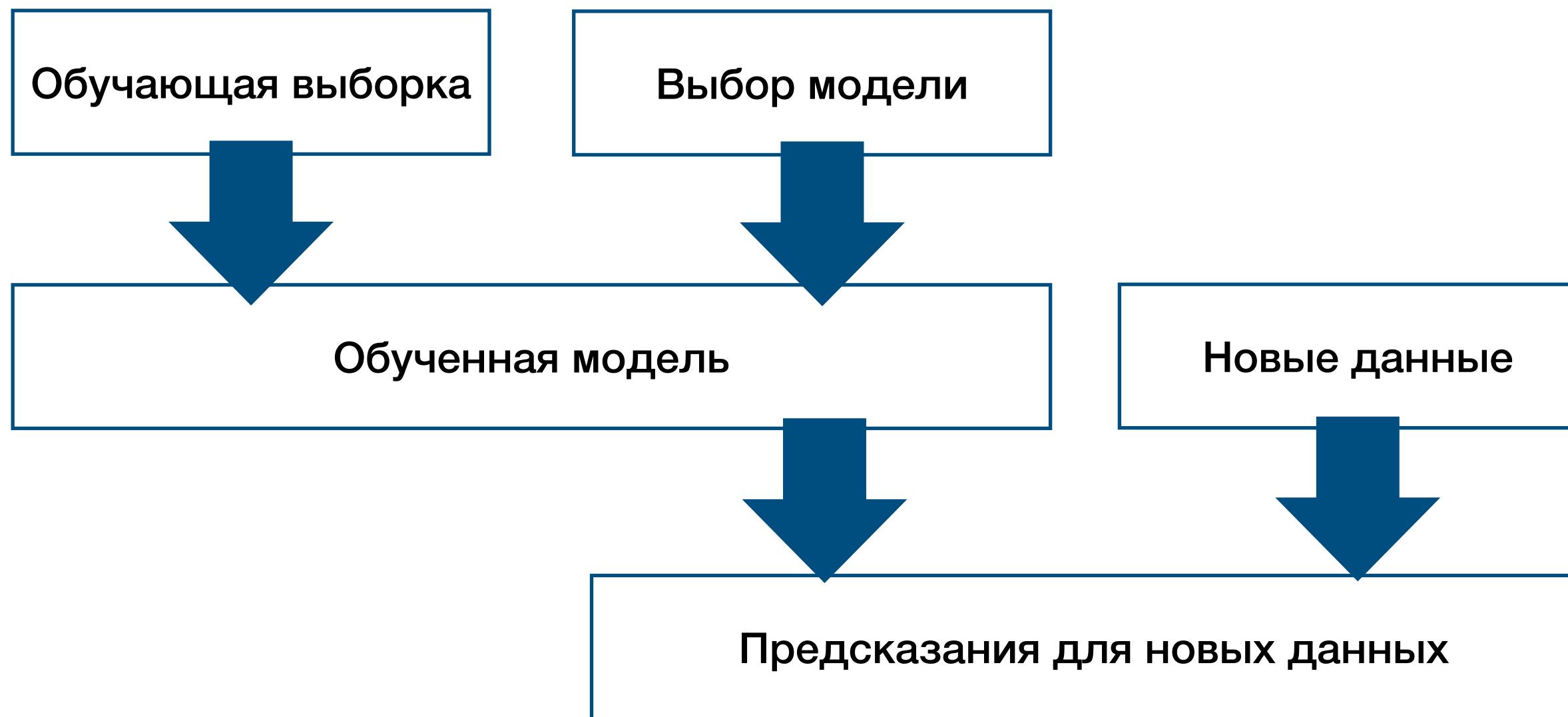


Генетический алгоритм



Постановка задачи машинного обучения

Задача: восстановить сложную зависимость по конечному числу примеров



Обучающая выборка

Матрица «объекты–признаки»

Датасет с задержками рейсов.

	Month	DayofMonth	DayOfWeek	DepTime	UniqueCarrier	Origin	Dest	Distance	dep_delayed_15min
0	c-8	c-21	c-7	1934	AA	ATL	DFW	732	N
1	c-4	c-20	c-3	1548	US	PIT	MCO	834	N
2	c-9	c-2	c-5	1422	XE	RDU	CLE	416	N
3	c-11	c-25	c-6	1015	OO	DEN	MEM	872	N
4	c-10	c-7	c-6	1828	WN	MDW	OMA	423	Y

Источник: <https://www.transtats.bts.gov>

Обучающая выборка

Матрица «объекты–признаки»

Датасет с задержками рейсов.

	Month	DayofMonth	DayOfWeek	DepTime	UniqueCarrier	Origin	Dest	Distance	dep_delayed_15min
0	c-8	c-21	c-7	1934	AA	ATL	DFW	732	N
1	c-4	c-20	c-3	1548	US	PIT	MCO	834	N
2	c-9	c-2	c-5	1422	XE	RDU	CLE	416	N
3	c-11	c-25	c-6	1015	OO	DEN	MEM	872	N
4	c-10	c-7	c-6	1828	WN	MDW	OMA	423	Y

Признаки

Источник: <https://www.transtats.bts.gov>

Обучающая выборка

Матрица «объекты–признаки»

Датасет с задержками рейсов.

	Month	DayofMonth	DayOfWeek	DepTime	UniqueCarrier	Origin	Dest	Distance	dep_delayed_15min
0	c-8	c-21	c-7	1934	AA	ATL	DFW	732	N
1	c-4	c-20	c-3	1548	US	PIT	MCO	834	N
2	c-9	c-2	c-5	1422	XE	RDU	CLE	416	N
3	c-11	c-25	c-6	1015	OO	DEN	MEM	872	N
4	c-10	c-7	c-6	1828	WN	MDW	OMA	423	Y

Объекты (прецеденты)

Источник: <https://www.transtats.bts.gov>

Обучающая выборка

Матрица «объекты–признаки»

Датасет с задержками рейсов.

	Month	DayofMonth	DayOfWeek	DepTime	UniqueCarrier	Origin	Dest	Distance	dep_delayed_15min
0	c-8	c-21	c-7	1934	AA	ATL	DFW	732	N
1	c-4	c-20	c-3	1548	US	PIT	MCO	834	N
2	c-9	c-2	c-5	1422	XE	RDU	CLE	416	N
3	c-11	c-25	c-6	1015	OO	DEN	MEM	872	N
4	c-10	c-7	c-6	1828	WN	MDW	OMA	423	Y

Целевая переменная

Источник: <https://www.transtats.bts.gov>

Признаки

Признаковое описание объекта - Вектор:

$$x_i = \{d_1, d_2, d_3, \dots, d_n\}$$

Множество значений признака

$$d_j \in D_j$$

Бинарные признаки

$$D_j = \{0, 1\}$$

В нашем примере:
Целевая переменная

Категориальные признаки

В нашем примере:
Локация отправления
Локация прибытия

Вещественные признаки

$$D_j = \mathbb{R}^m$$

В нашем примере:
Расстояние

Формальная постановка задачи

Дана обучающая выборка (объекты независимы):

$$X_m = \{ (x_1, y_1), \dots, (x_m, y_m) \}$$

Для задачи регрессии - Целевая переменная задана вещественным числом

$$(x_1, y_1) \in \mathbb{R}^m \times \mathbb{Y}, \mathbb{Y} = \mathbb{R}$$

Для задачи классификации - Целевая переменная задана конечным числом меток

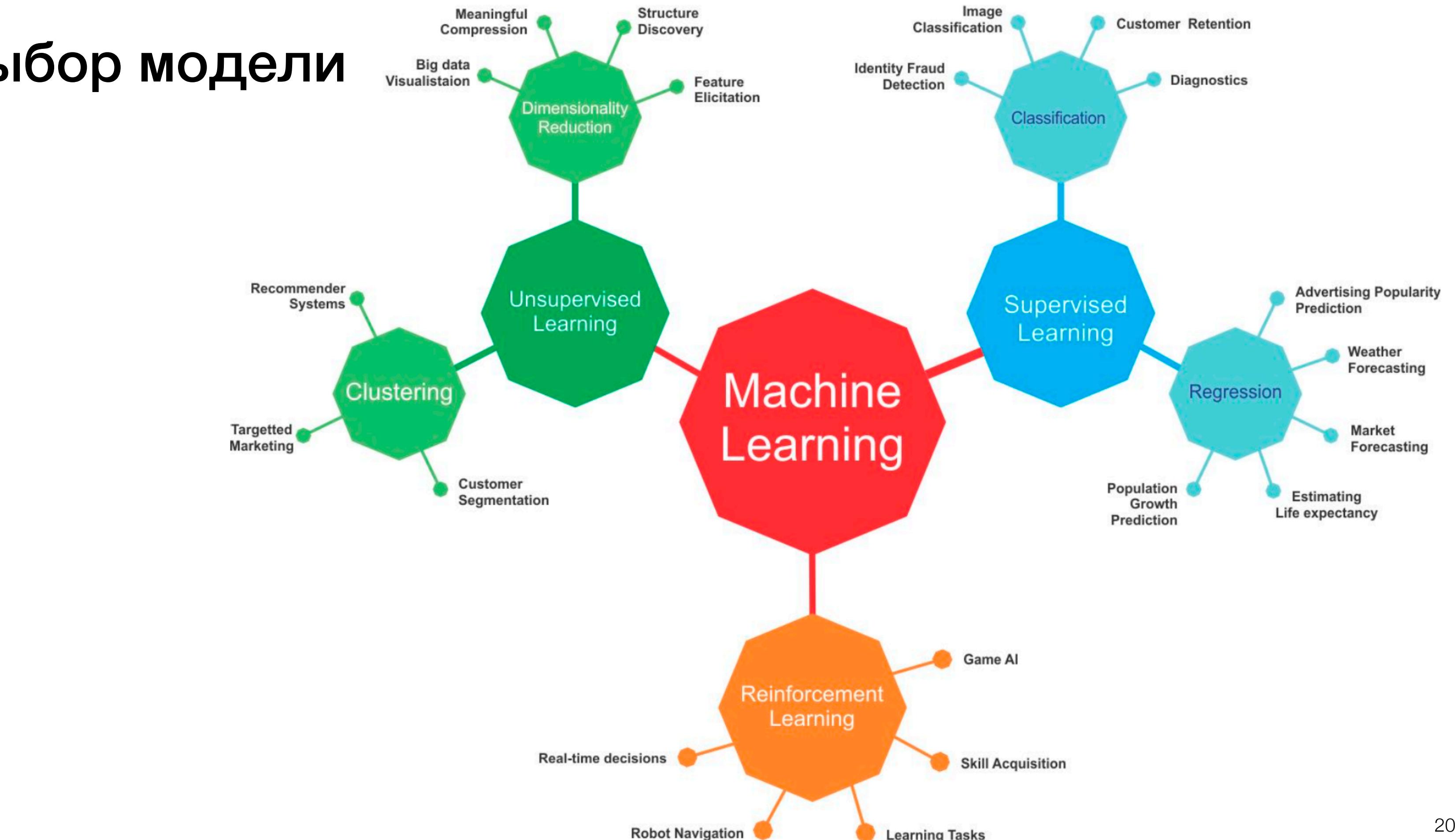
$$(x_1, y_1) \in \mathbb{R}^m \times \mathbb{Y}, \mathbb{Y} = \{-1; 1\}$$

Задать такую функцию $f(x)$ от вектора признаков x , которое
выдаст ответ для любого возможного наблюдения x

$$f(x): \mathbb{X} \rightarrow \mathbb{Y}$$

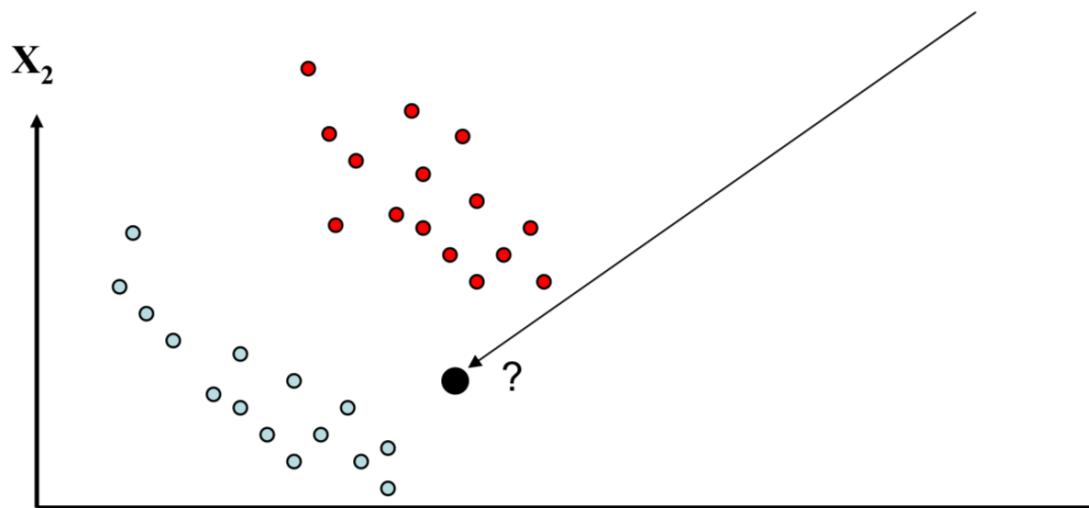
Основная гипотеза МО: Схожим объектам соответствуют схожие объекты

Выбор модели

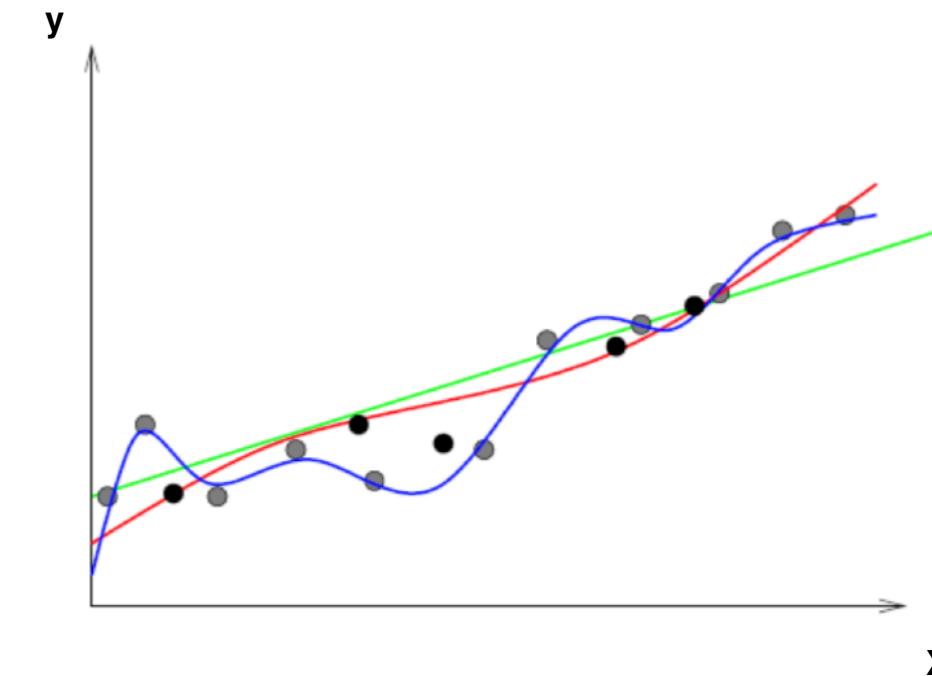


Формальная постановка задачи

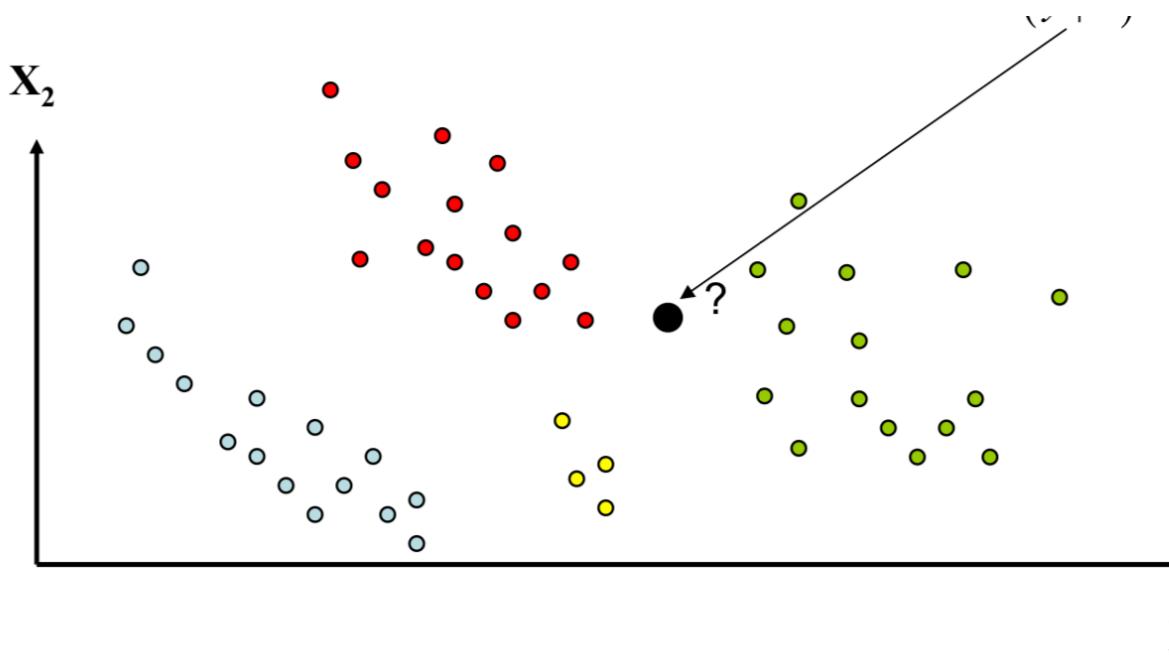
Классификация



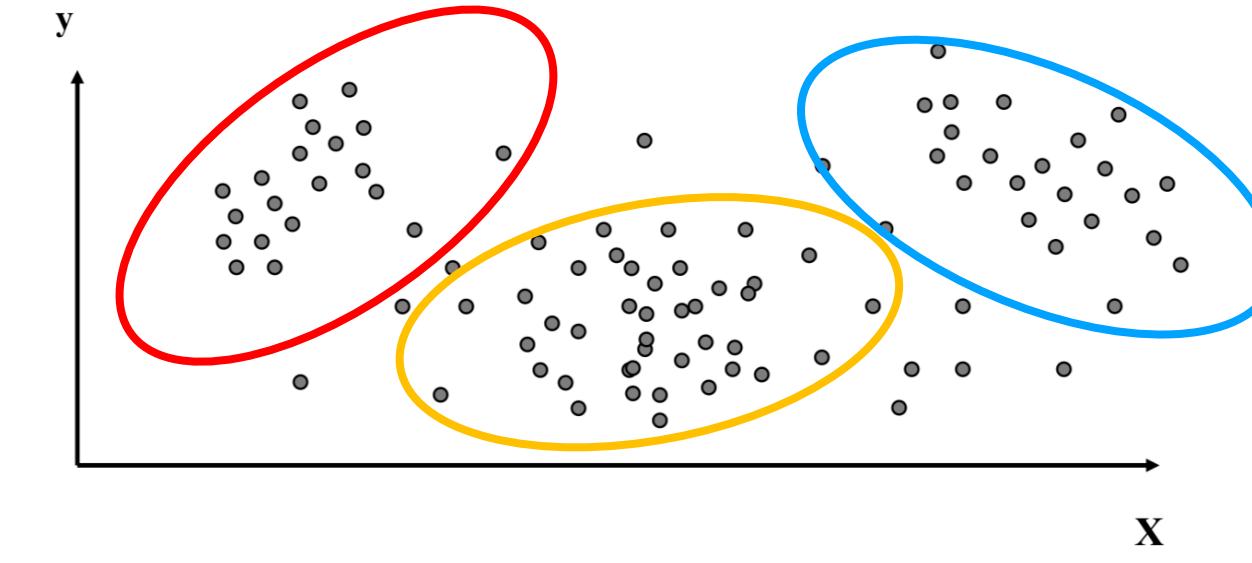
Регрессия



Классификация - многокласс



Кластеризация



Метрики качества в задачах регрессии

Средняя квадратичная ошибка
(*Mean Squared Error, MSE*):

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

Предсказание модели для
объекта x_i

Средняя абсолютная ошибка
(*Mean Absolute Error, MAE*):

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

Значение целевой
переменной для объекта
 x_i

Метрики качества в задачах регрессии

Коэффициент детерминации (R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^l (f(x_i) - y_i)^2}{\sum_{i=1}^l (y_i - \bar{y})^2}, \text{ где } \bar{y} = \sum_{i=1}^l y_i$$

Другие полезные метрики:

[Квантильная регрессия](#)

[Mean Absolute Percentage Error](#)

Метрики качества в задачах классификации

Матрица ошибок (confusion matrix):

		Actual class	
		Yes	No
Predicted class	Yes	True Positive (TP)	False Positive (FP)
	No	False Negative (FN)	True Negative (TN)

Метрики качества в задачах классификации

Матрица ошибок (*confusion matrix*):

		Actual class	
		Yes	No
Predicted class	Yes	90	20
	No	10	50

Выборка: Всего 170
Положительного класса 100
Отрицательного класса 70

Прогноз:
Положительного класса 110
Отрицательного класса 60

Метрики качества в задачах классификации

Матрица ошибок (confusion matrix):

		Actual class	
		Yes	No
Predicted class	Yes	True Positive (TP)	False Positive (FP)
	No	False Negative (FN)	True Negative (TN)

Доля правильных ответов (accuracy):

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Метрики качества в задачах классификации

Матрица ошибок (*confusion matrix*):

		Actual class	
		Yes	No
Predicted class	Yes	True Positive (TP)	False Positive (FP)
	No	False Negative (FN)	True Negative (TN)

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

		Actual class	
		Yes	No
Predicted class	Yes	90	20
	No	10	50

Метрики качества в задачах классификации

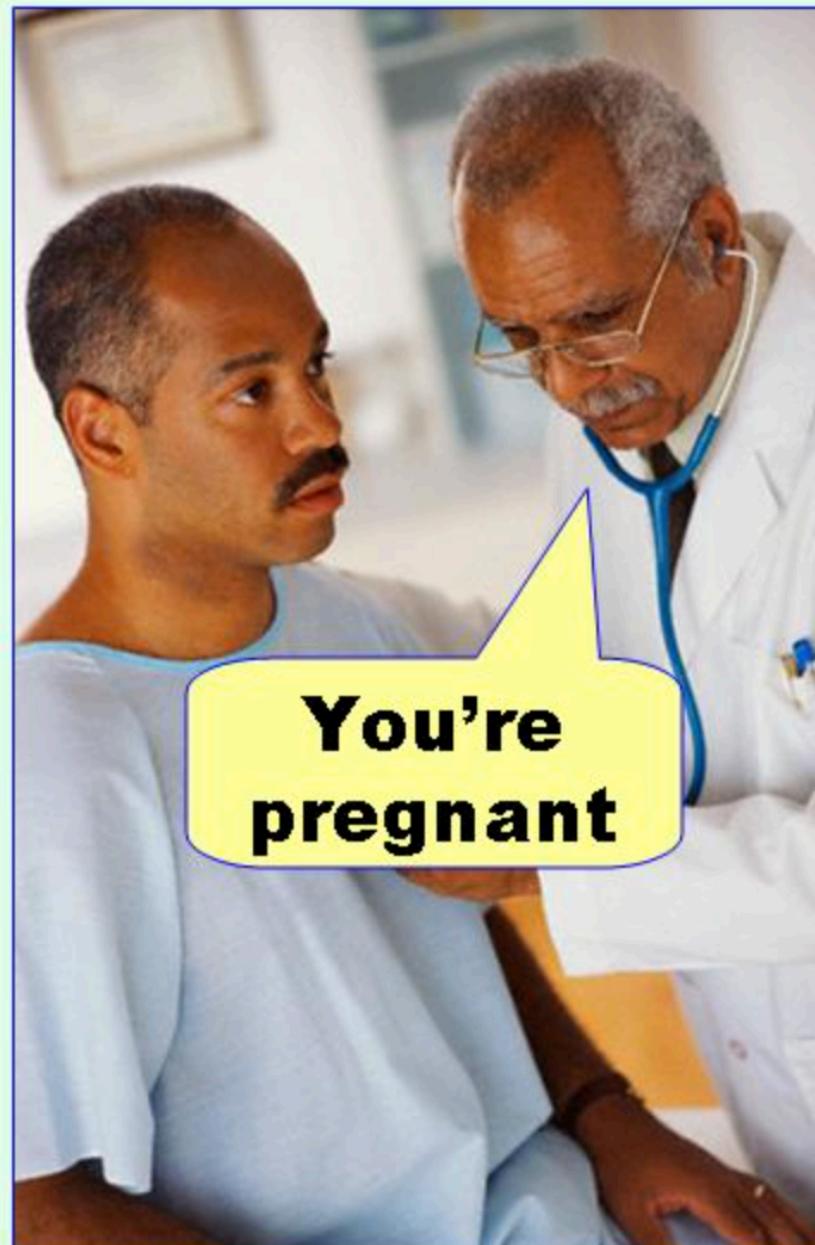
Матрица ошибок (*confusion matrix*):

		Actual class		
		Yes	No	
Predicted class	Yes	True Positive (TP)	False Positive (FP)	Ошибка I-ого рода
	No	False Negative (FN)	True Negative (TN)	Ошибка II-ого рода

Метрики качества в задачах классификации

Type I error

(false positive)



Type II error

(false negative)



Метрики качества в задачах классификации

Матрица ошибок (*confusion matrix*):

		Actual class	
		Yes	No
Predicted class	Yes	True Positive (TP)	False Positive (FP)
	No	False Negative (FN)	True Negative (TN)

Точность (*precision*):

$$precision = \frac{TP}{TP + FP}$$

Доля объектов, предсказанных как положительные, действительно является положительными.

Полнота (*recall*):

$$recall = \frac{TP}{TP + FN}$$

Доля положительных объектов, которую выделил классификатор

Метрики качества в задачах классификации

Матрица ошибок (*confusion matrix*):

		Actual class	
		Yes	No
Predicted class	Yes	True Positive (TP)	False Positive (FP)
	No	False Negative (FN)	True Negative (TN)

F -мера:

$$F_\beta = (1 + \beta^2) \frac{precision \times recall}{\beta^2 precision + recall}$$

$$F = \frac{2 \times precision \times recall}{precision + recall}$$

Метрики качества в задачах классификации

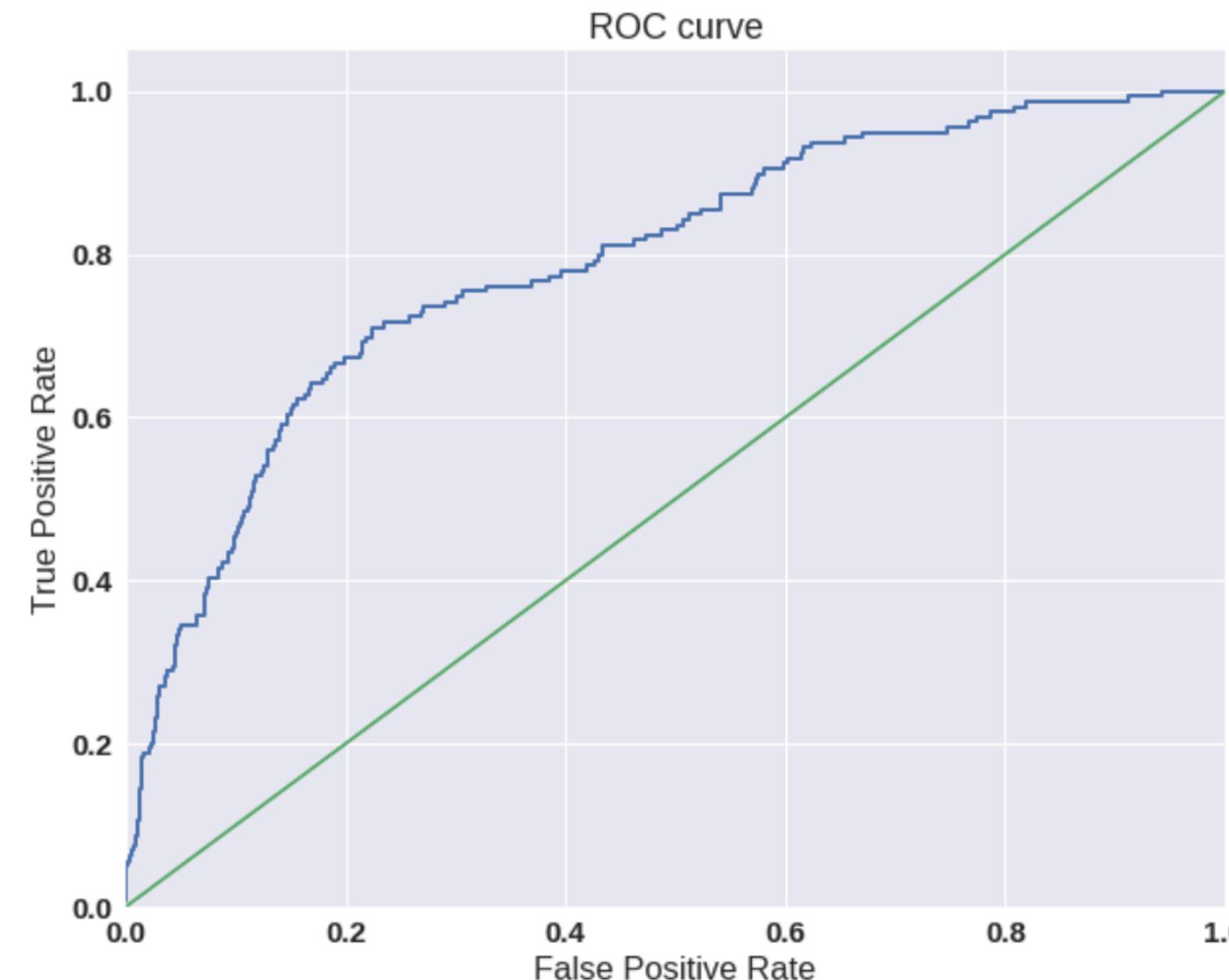
ROC AUC

или площадь (*Area Under Curve*) под кривой ошибок (*Receiver Operating Characteristic curve*).

Кривая ошибок (*Receiver Operating Characteristic curve*):

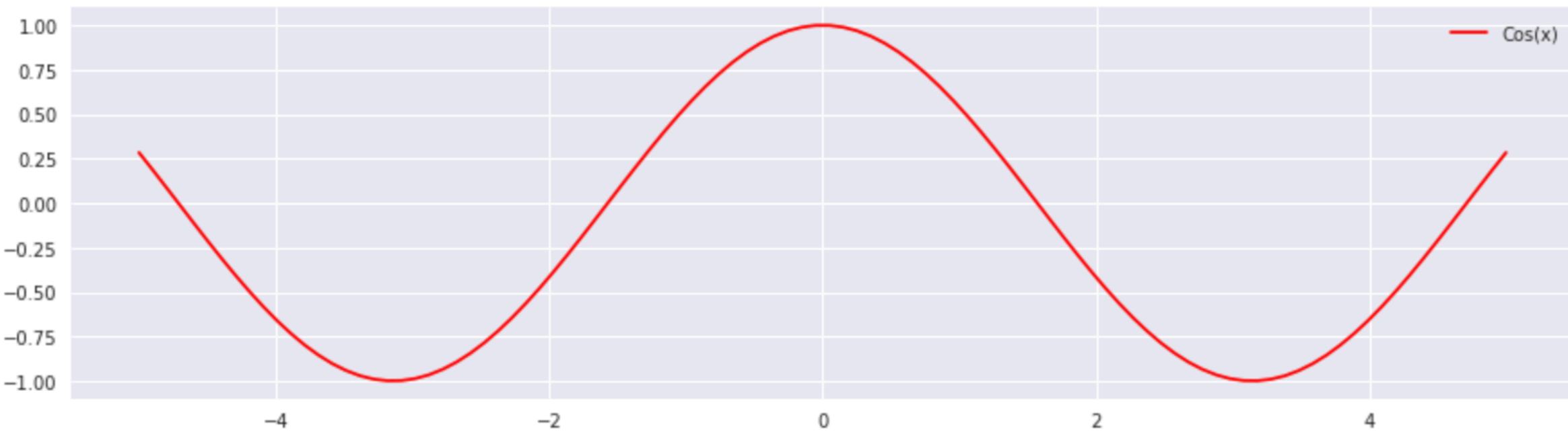
$$TPR = \frac{TP}{TP + FN} = recall$$

$$FPR = \frac{FP}{FP + TN}$$



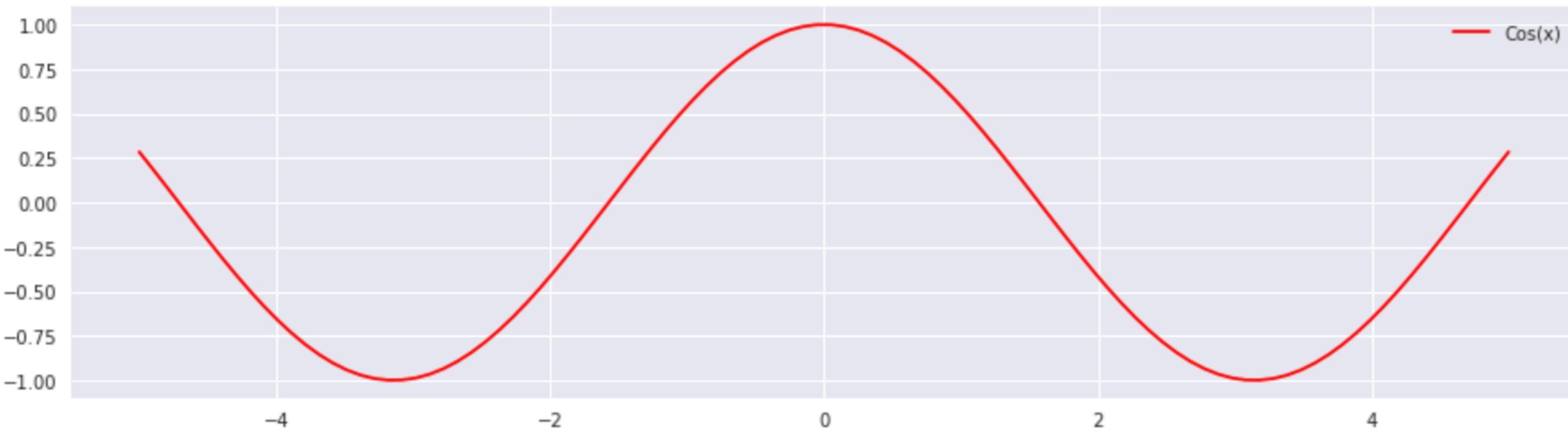
Мотивация валидации

$$y = \cos(x), x \in [-5, 5]$$

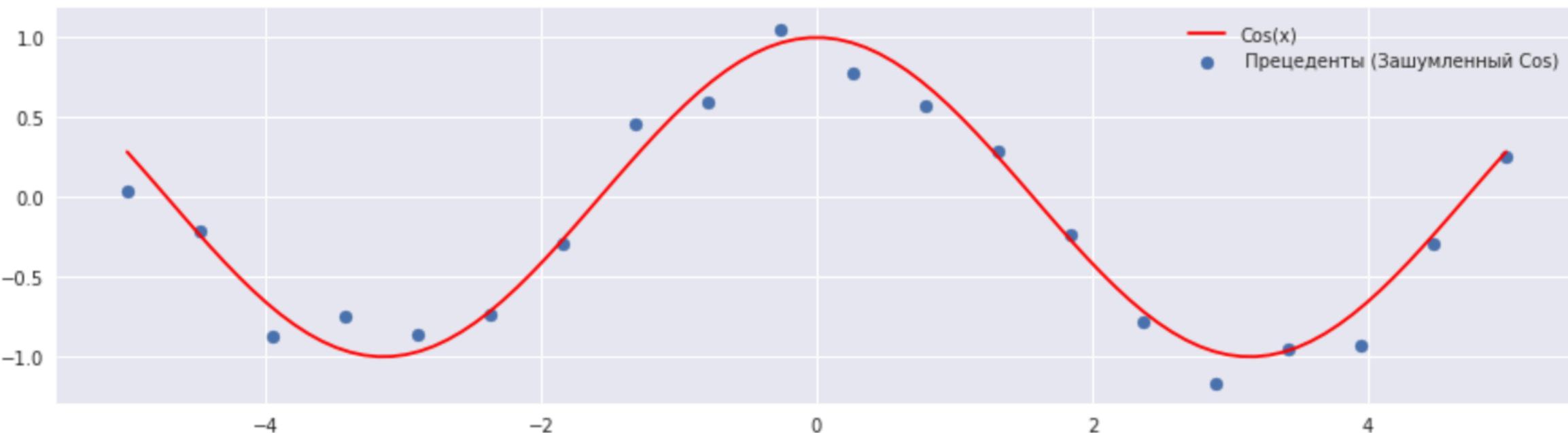


Мотивация валидации

$$y = \cos(x), x \in [-5, 5]$$

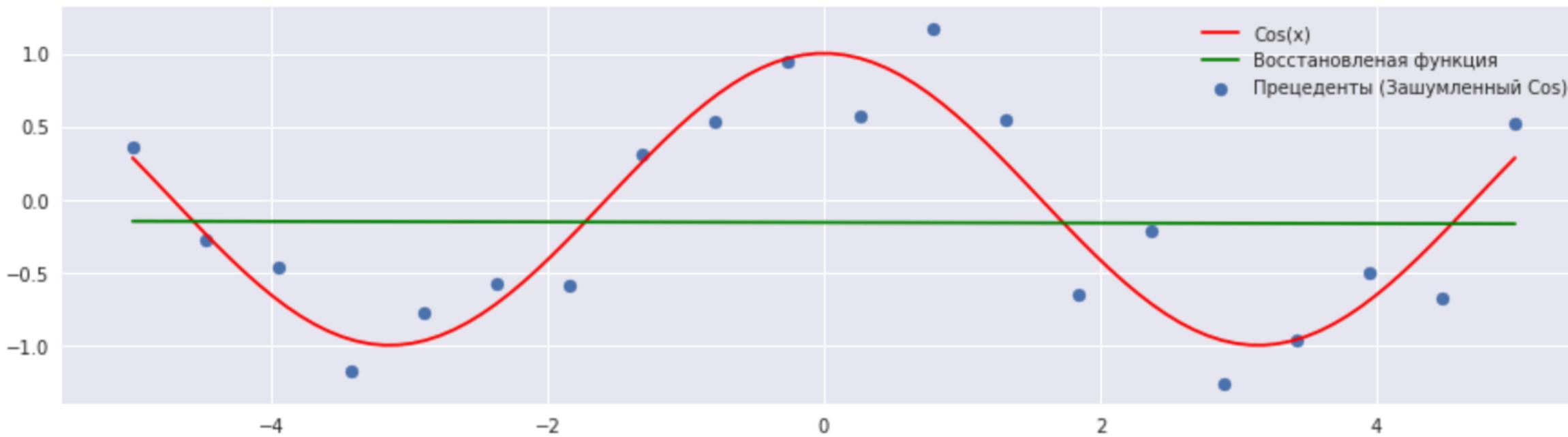


$$y = \cos(x) + \varepsilon, \text{ где } \varepsilon = \mathcal{N}\left(0, \frac{1}{2}\right), x \in [-5, 5]$$



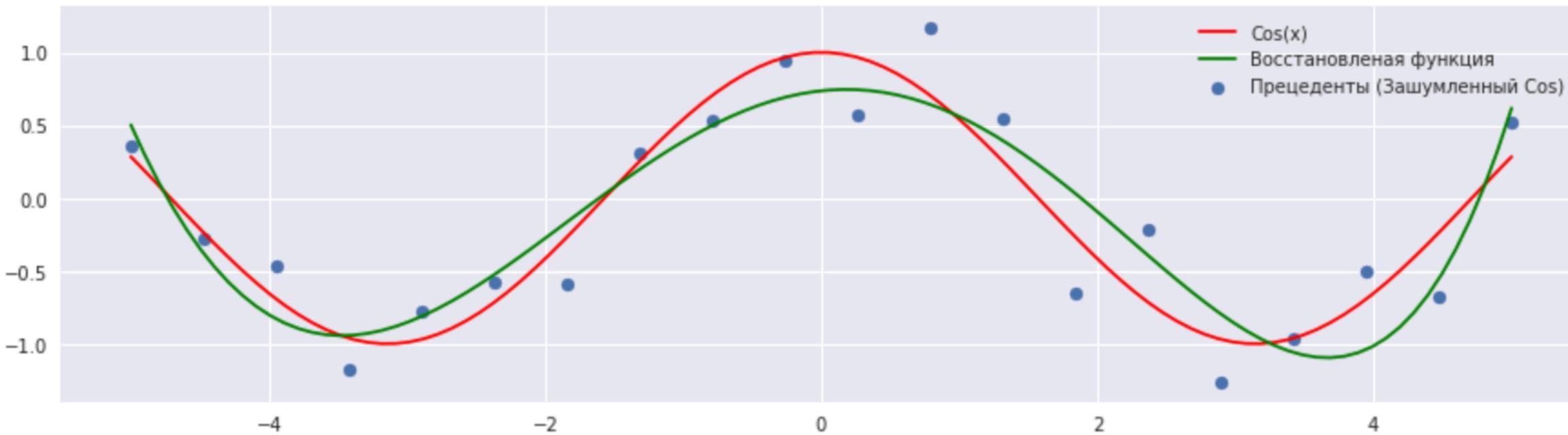
Мотивация валидации

Восстановим зависимость линейной функцией



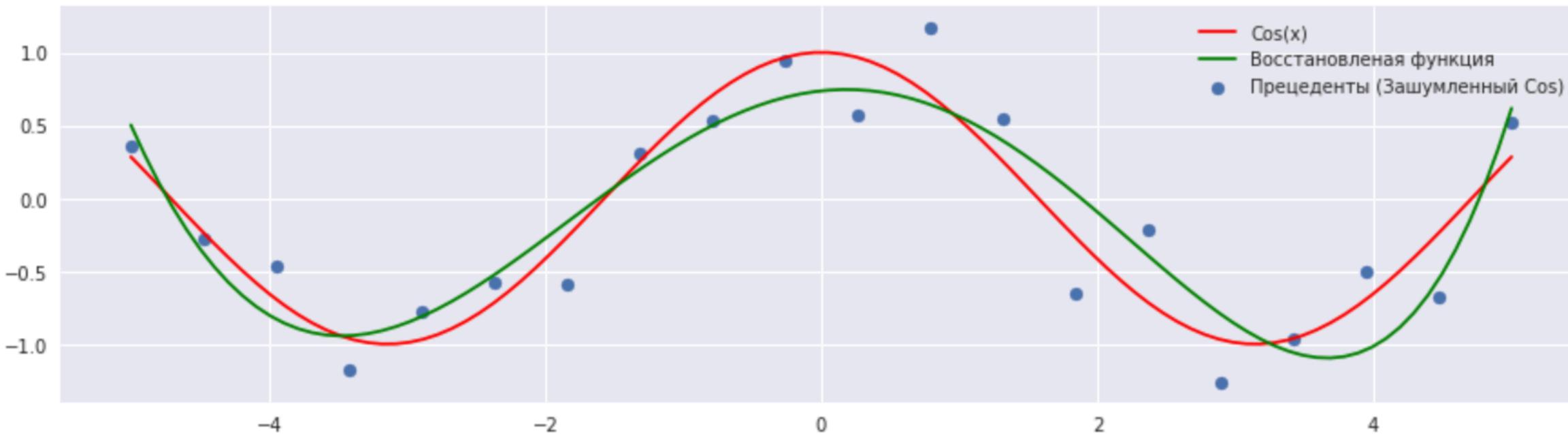
Мотивация валидации

Восстановим зависимость с помощью полинома 5-ого порядка

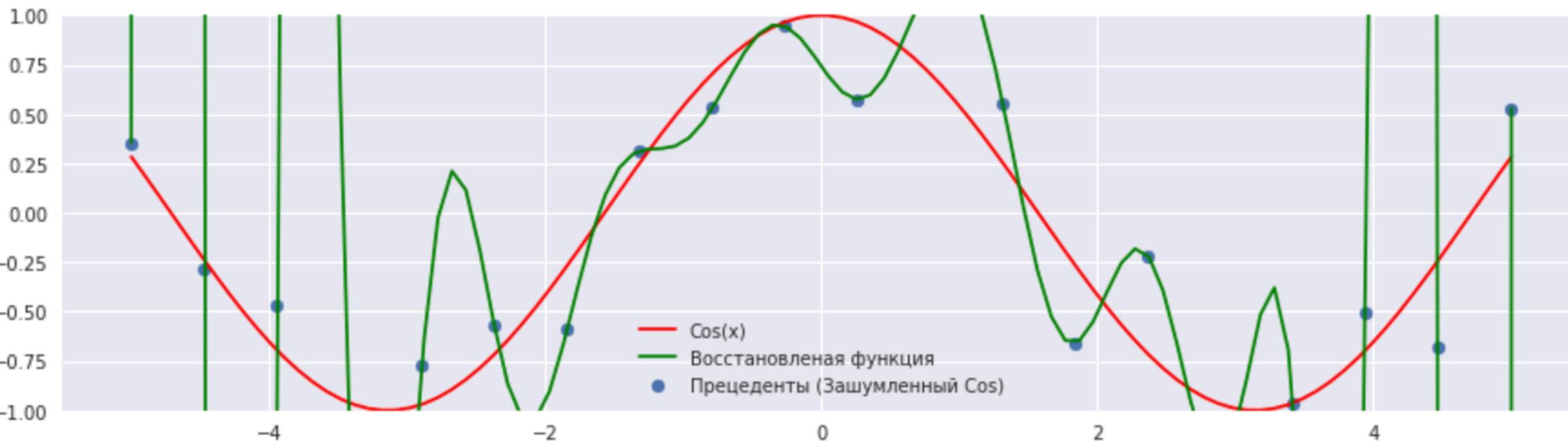


Мотивация валидации

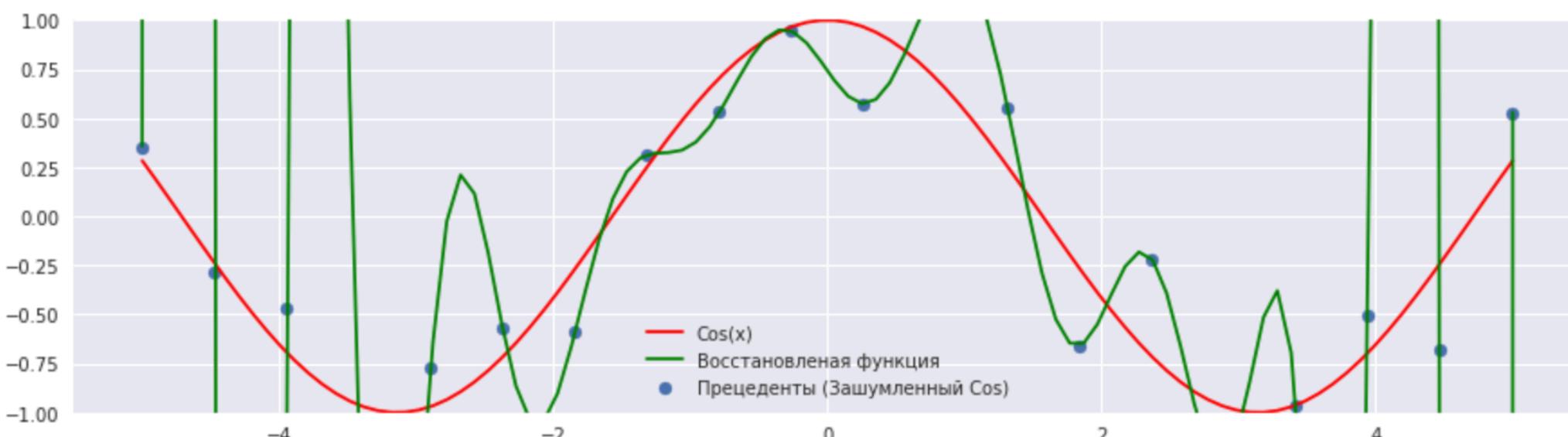
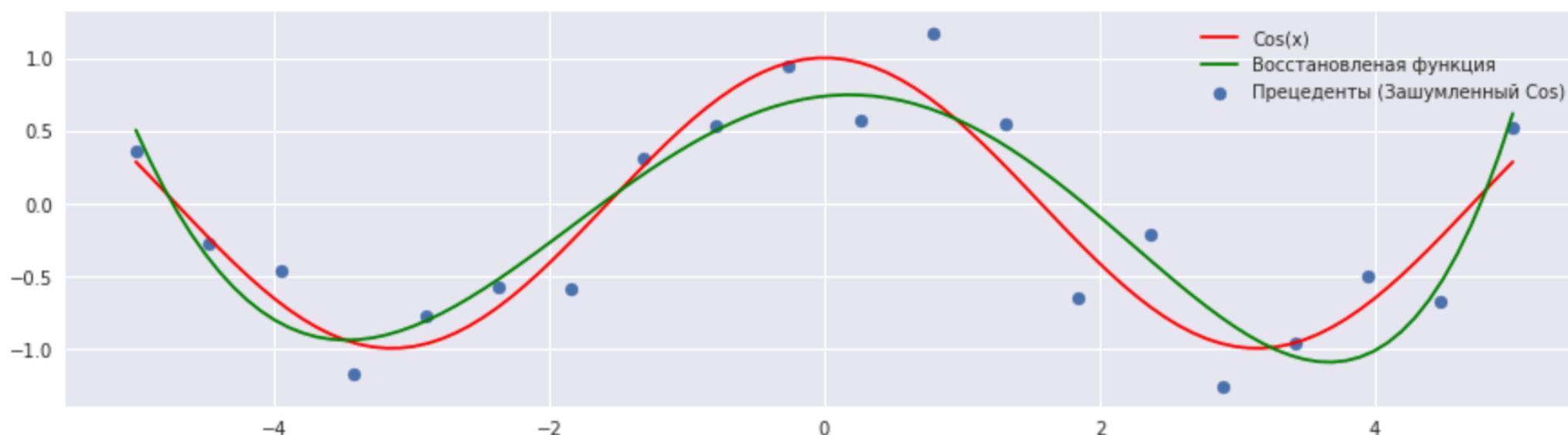
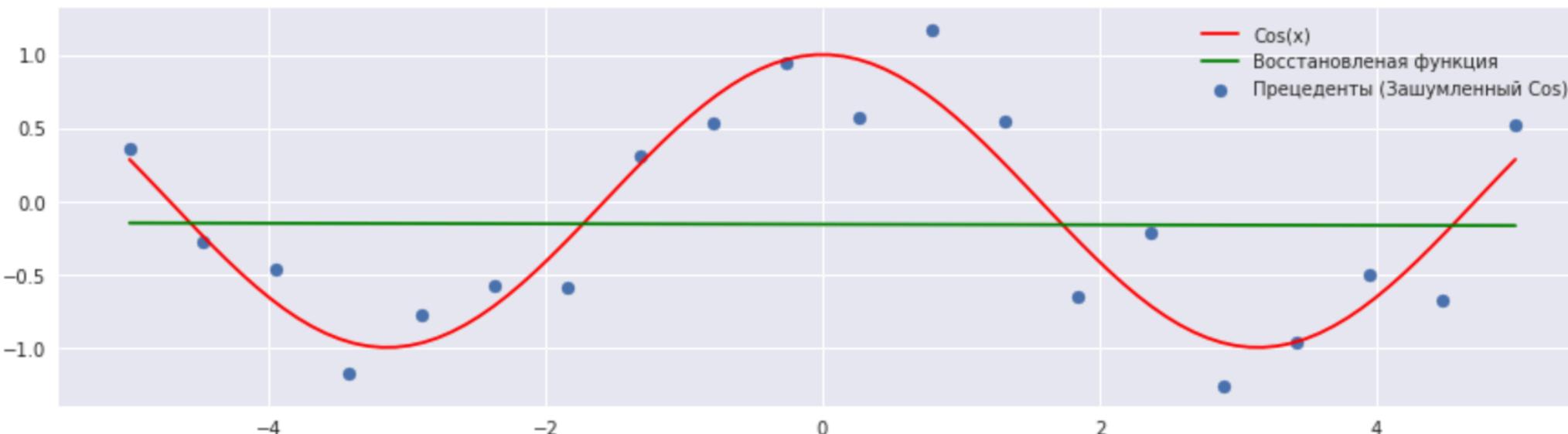
Восстановим зависимость с помощью полинома 5-ого порядка



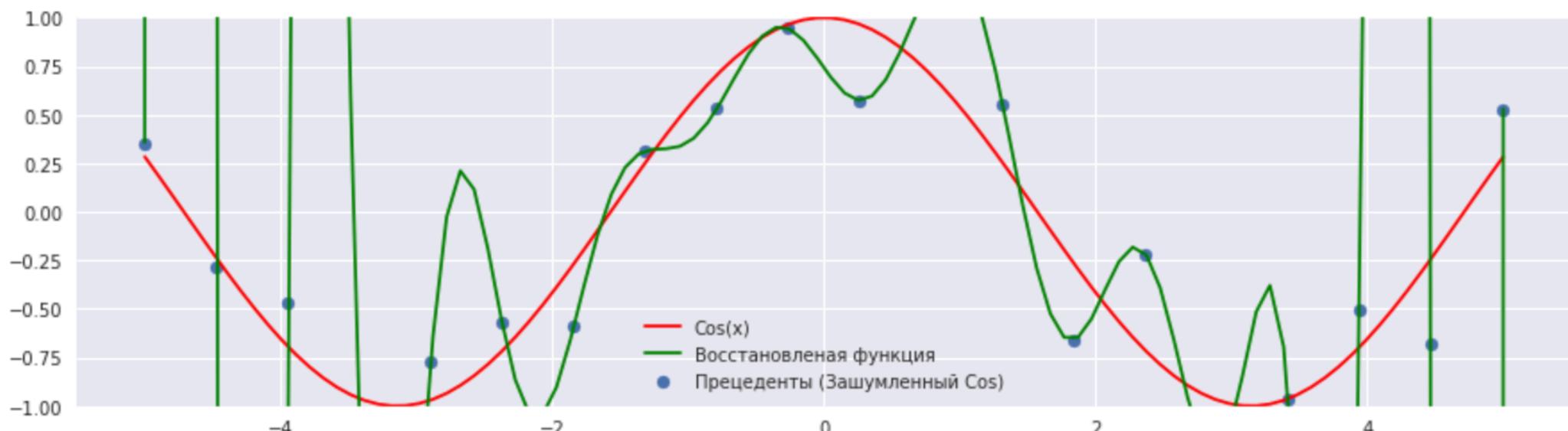
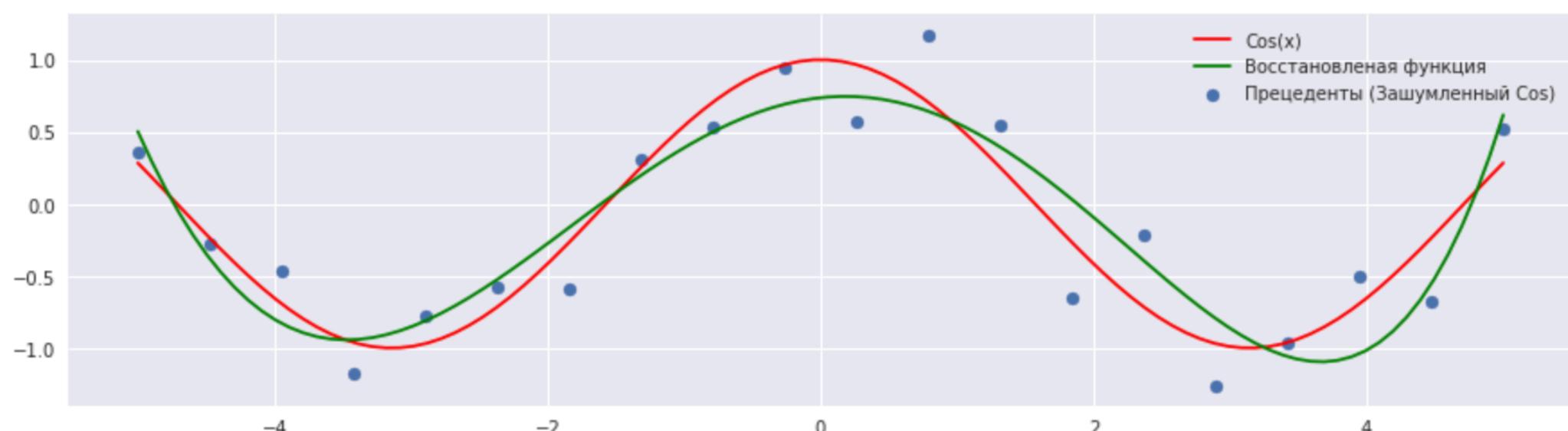
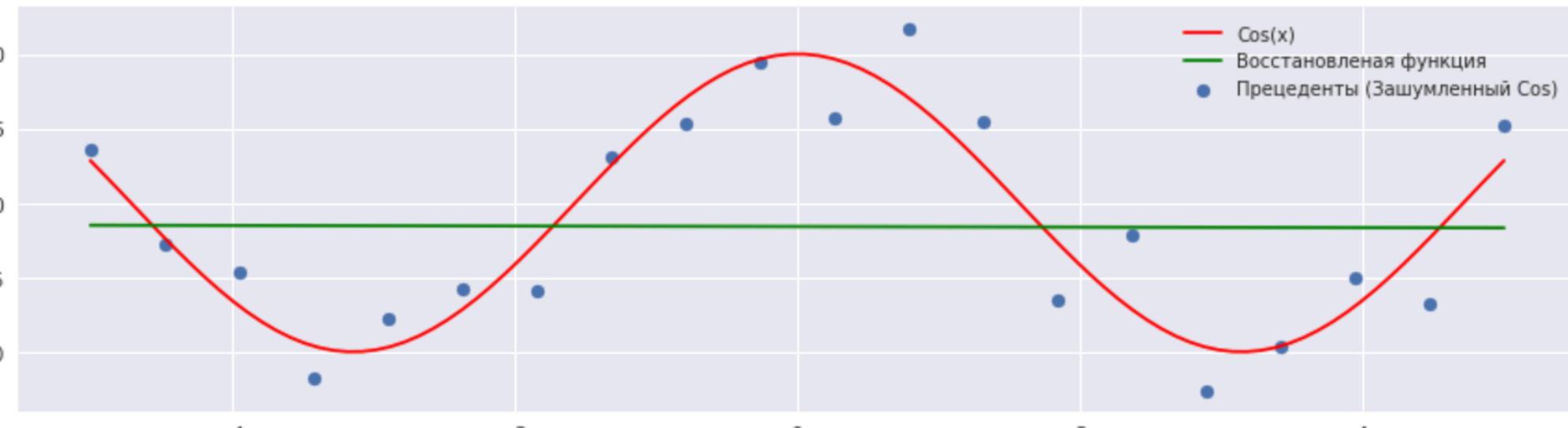
Восстановим зависимость с помощью полинома 11-ого порядка



Мотивация валидации

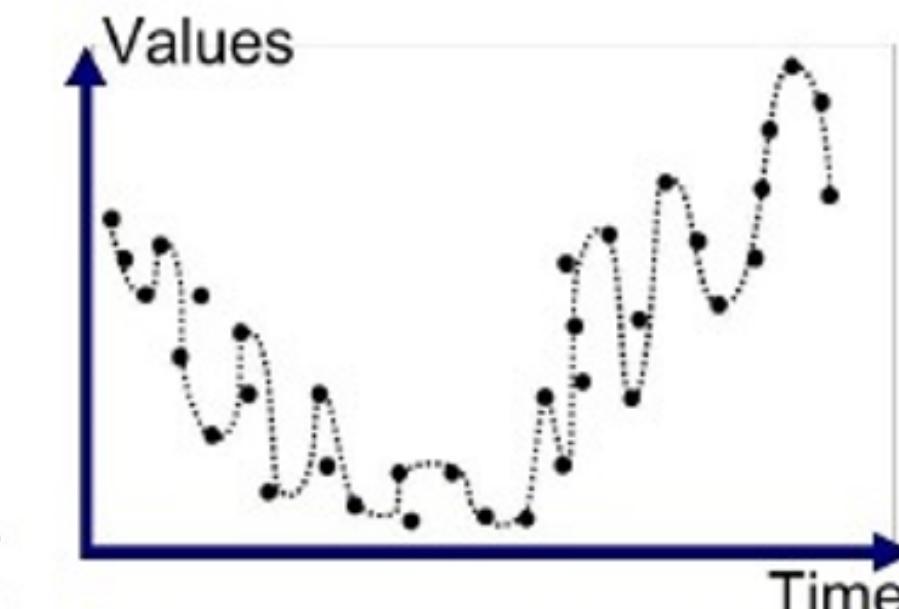
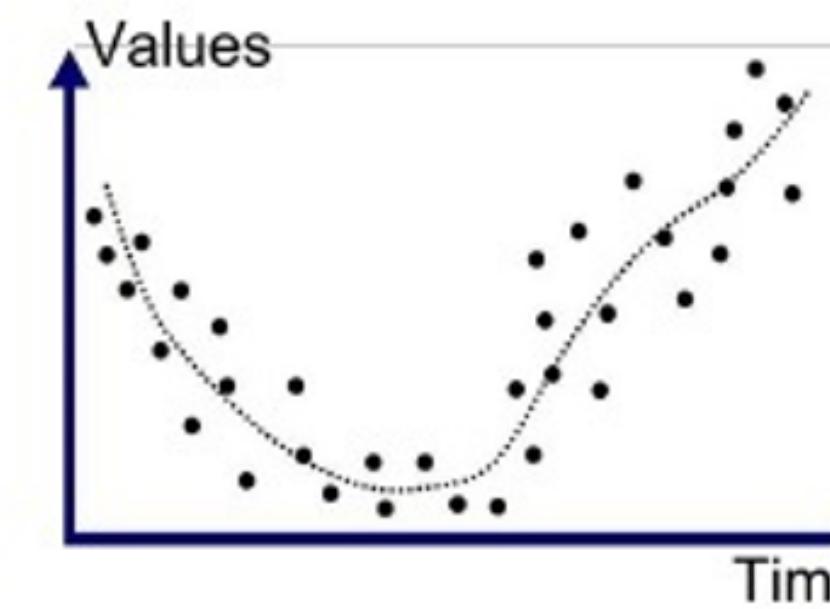
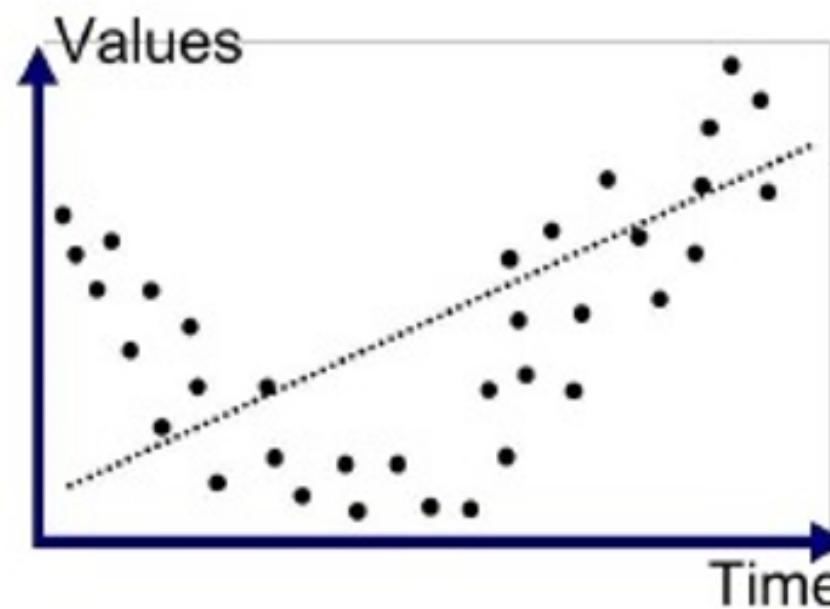


Мотивация валидации



	MSE	MAE	R2
Линейная модель	0.472	0.586	0.0004
Полином 5-ой степени	0.047	0.179	0.9000
Полином 11-ой степени	0.000	0.000	1.0000

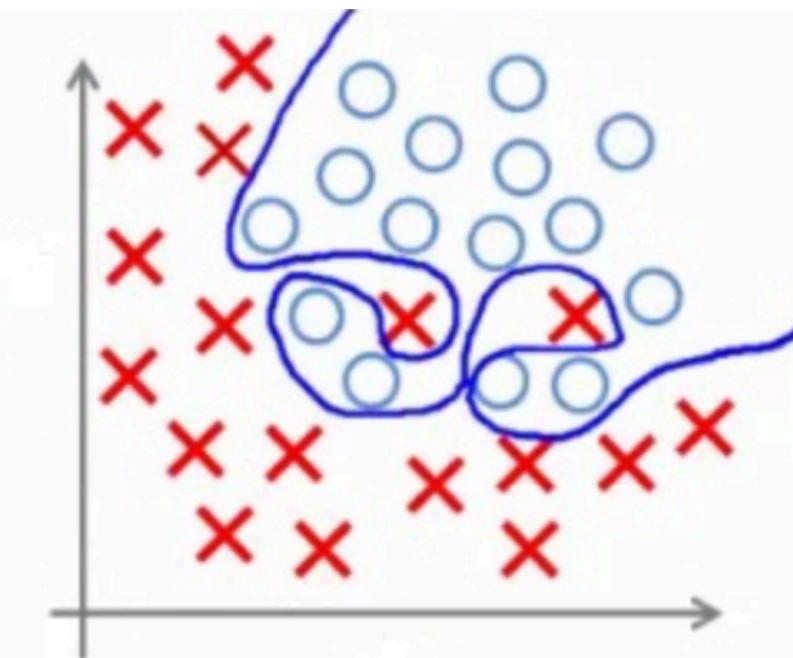
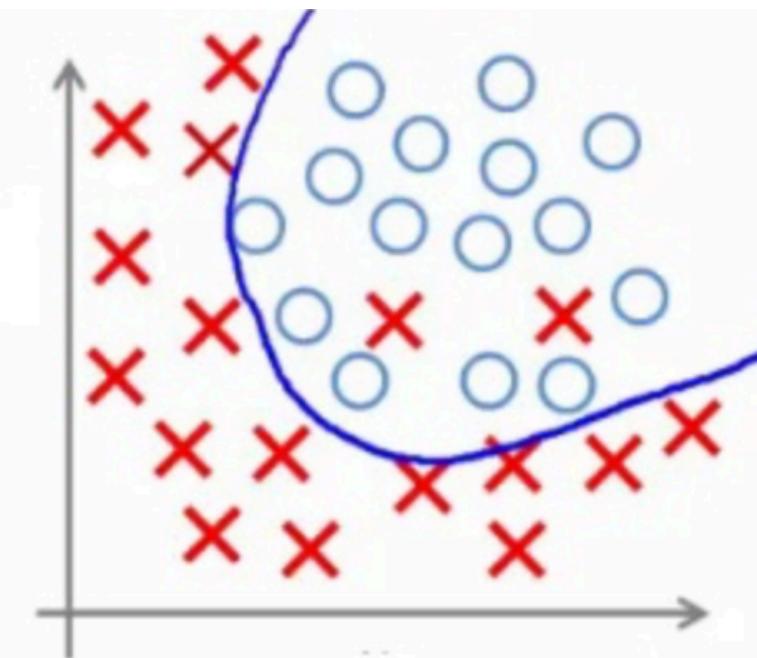
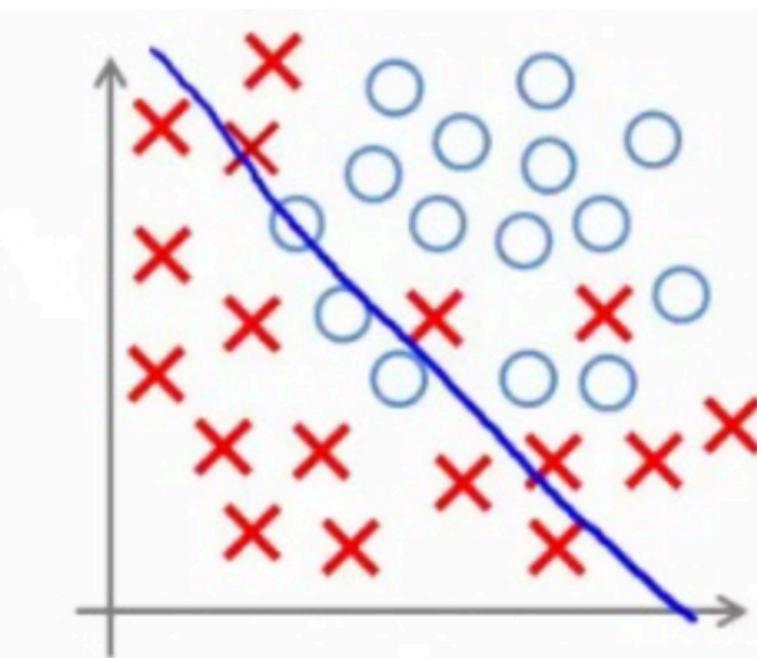
Мотивация валидации



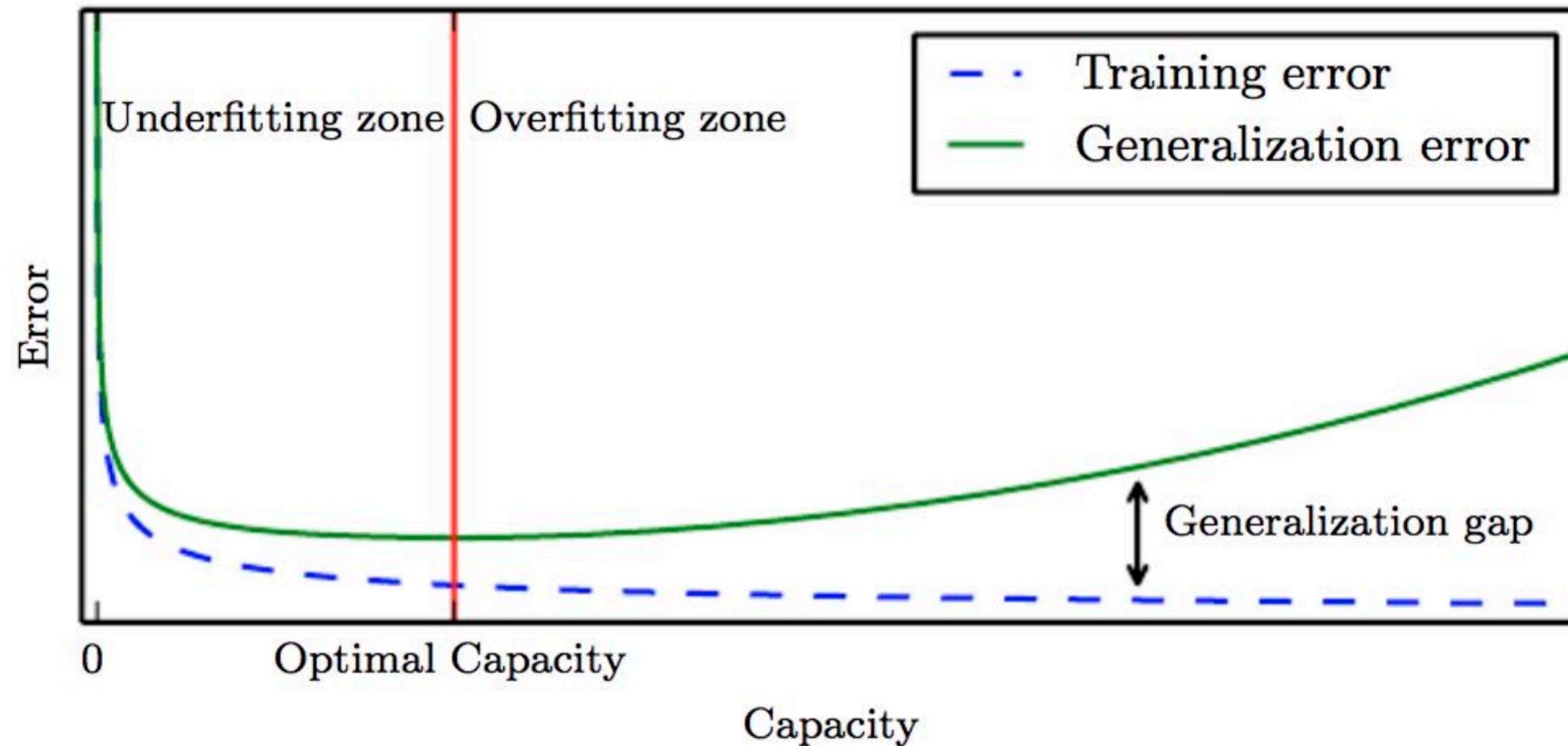
Underfitted

Good Fit/R robust

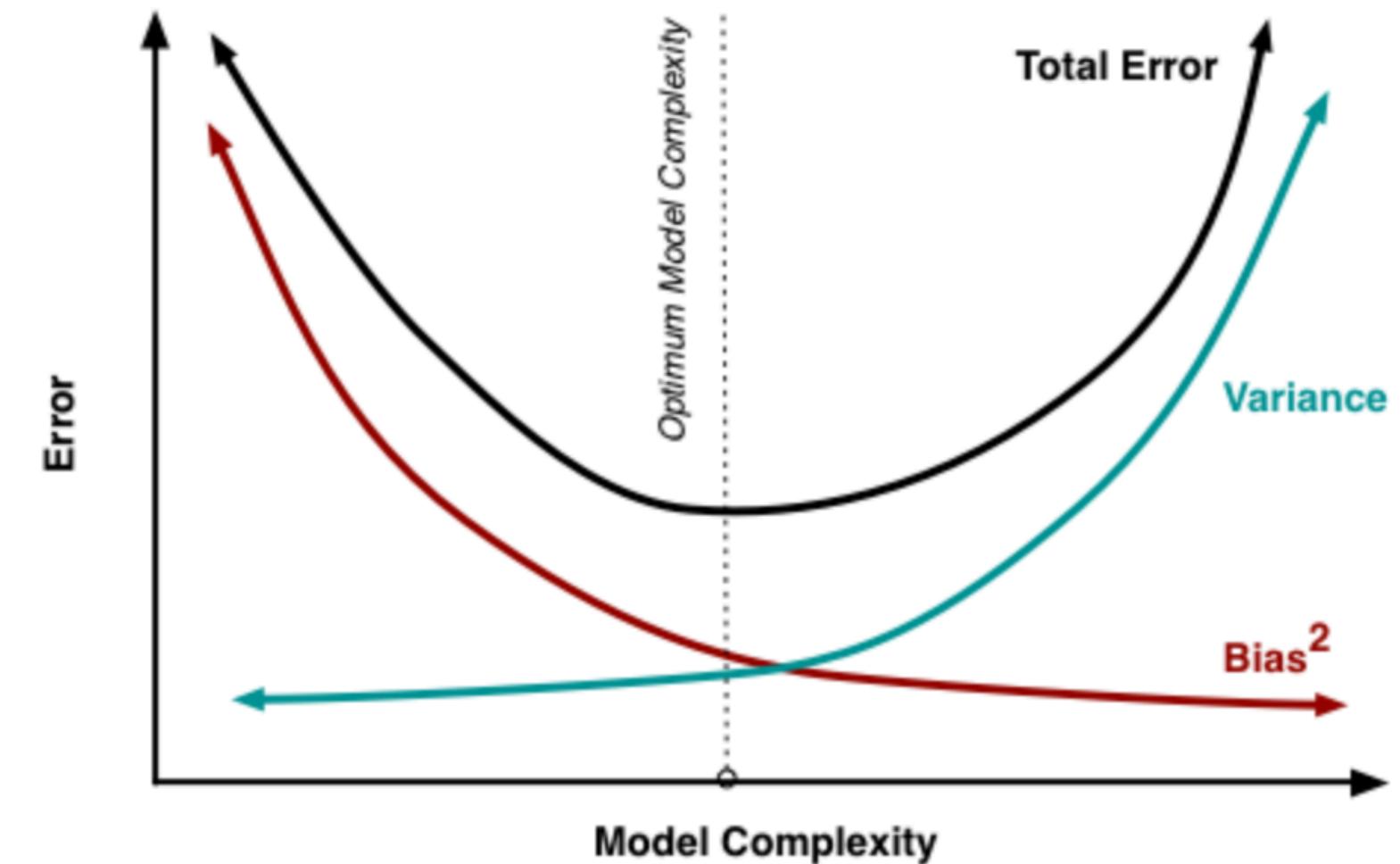
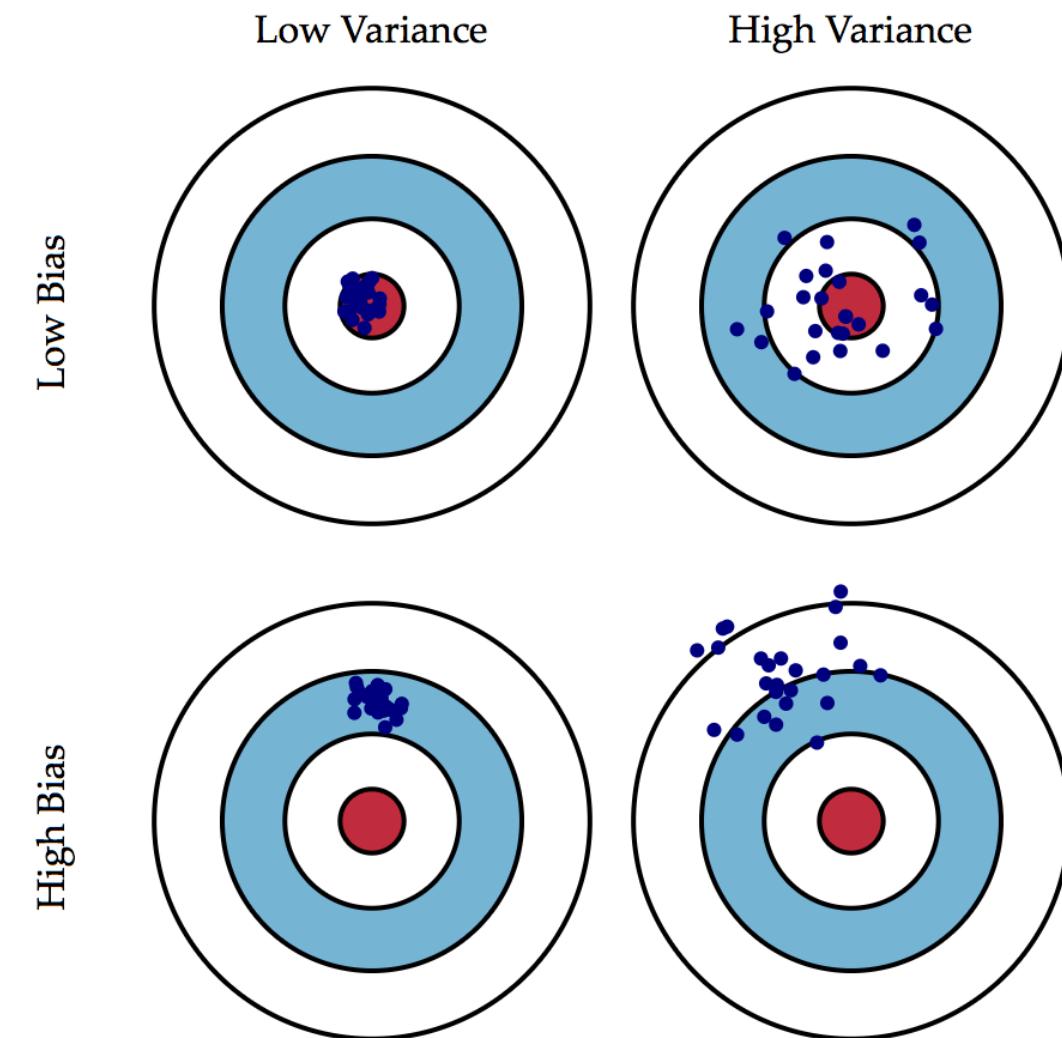
Overfitted



Мотивация валидации



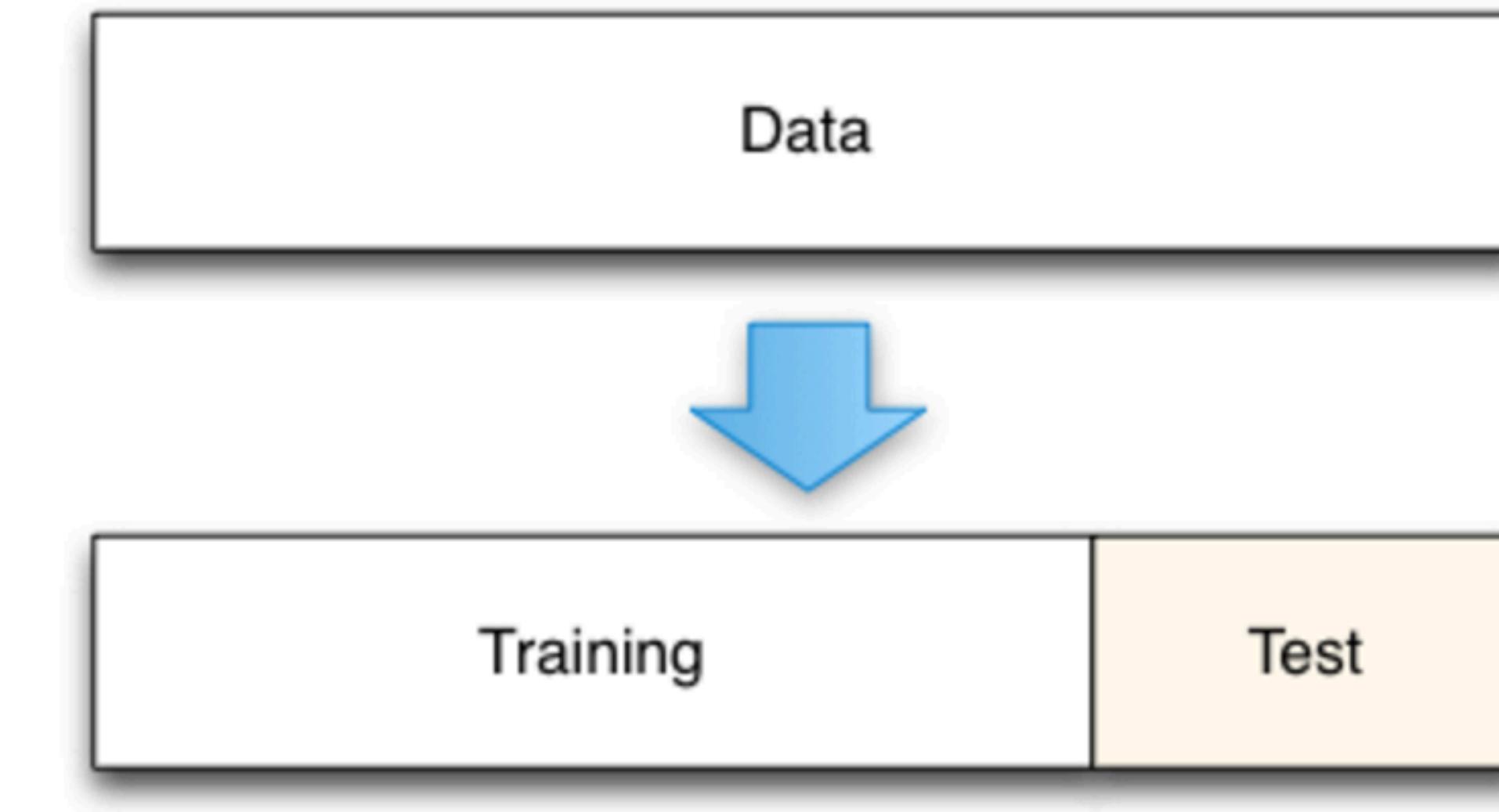
Bias and Variance tradeoff



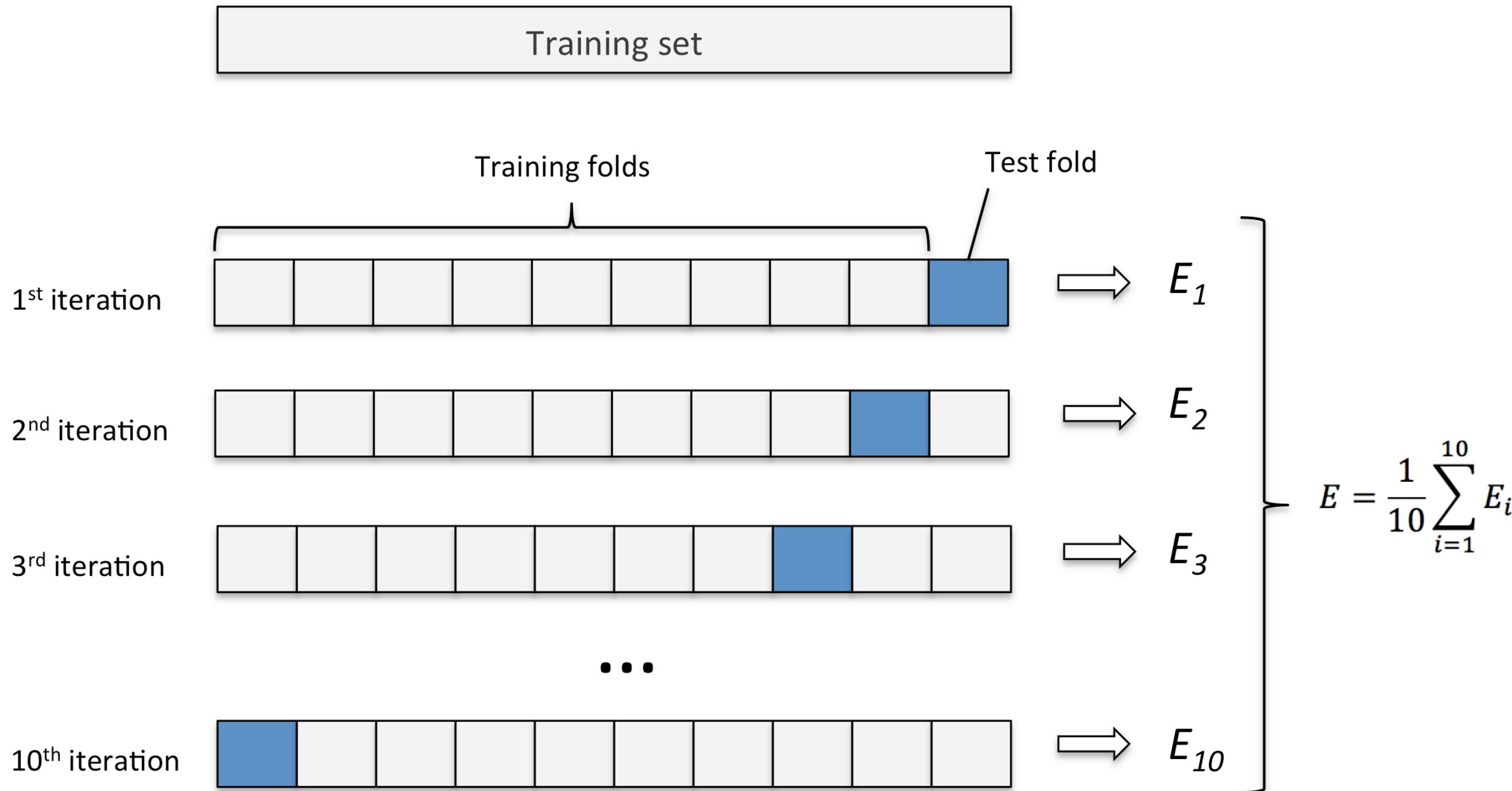
$$Err(x) = E[(Y - f(x))^2]$$

$$Err(x) = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

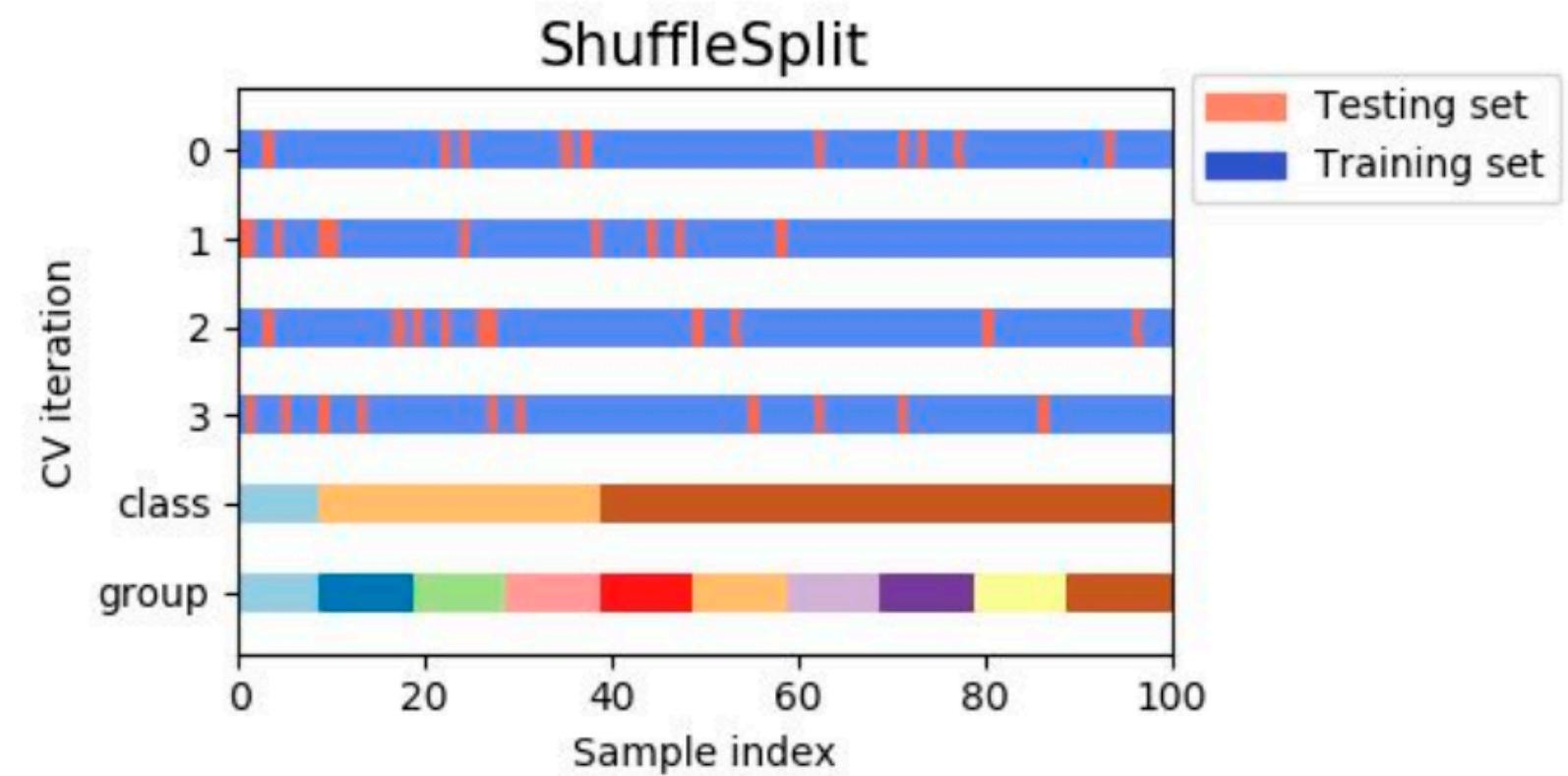
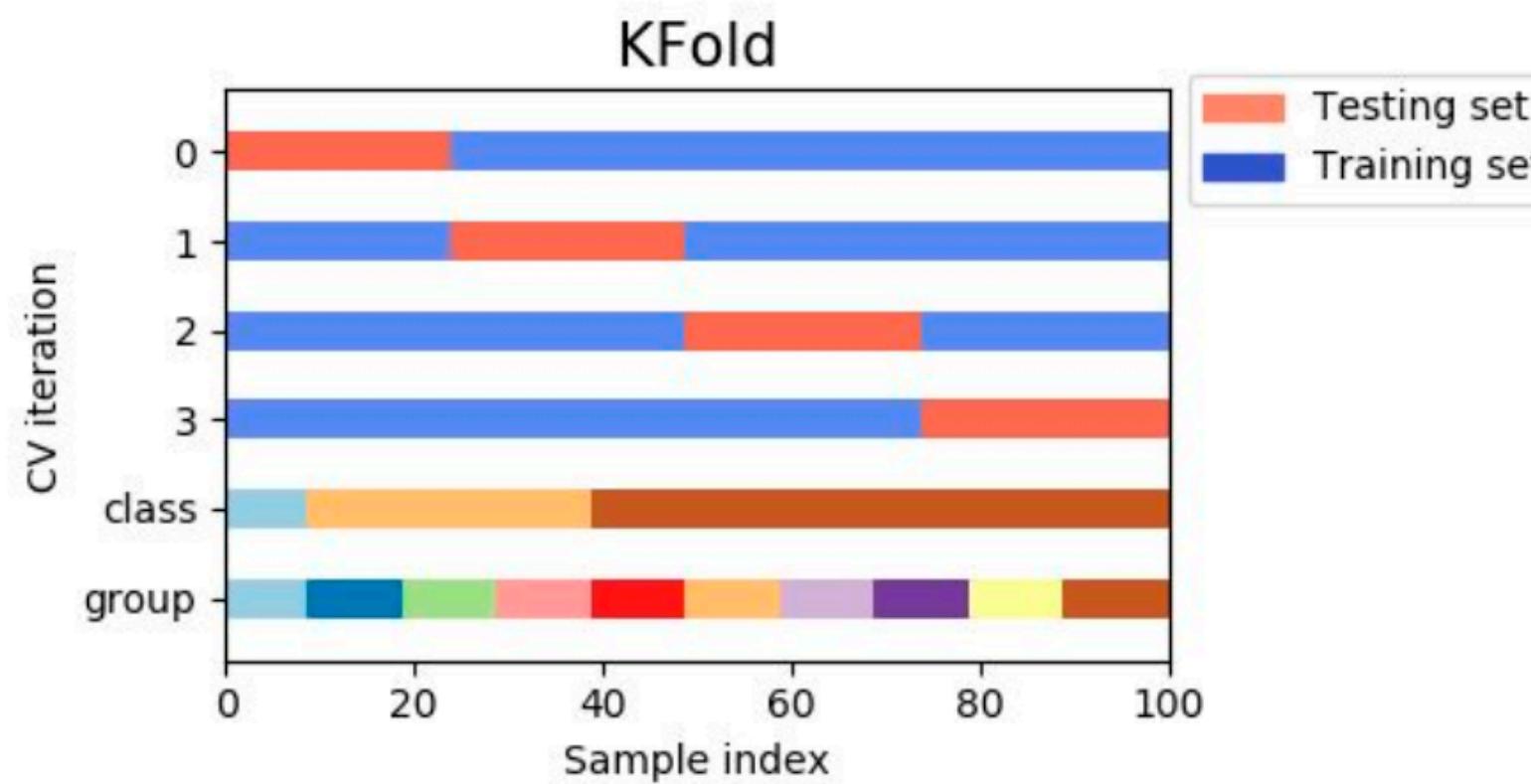
Стратегии валидации



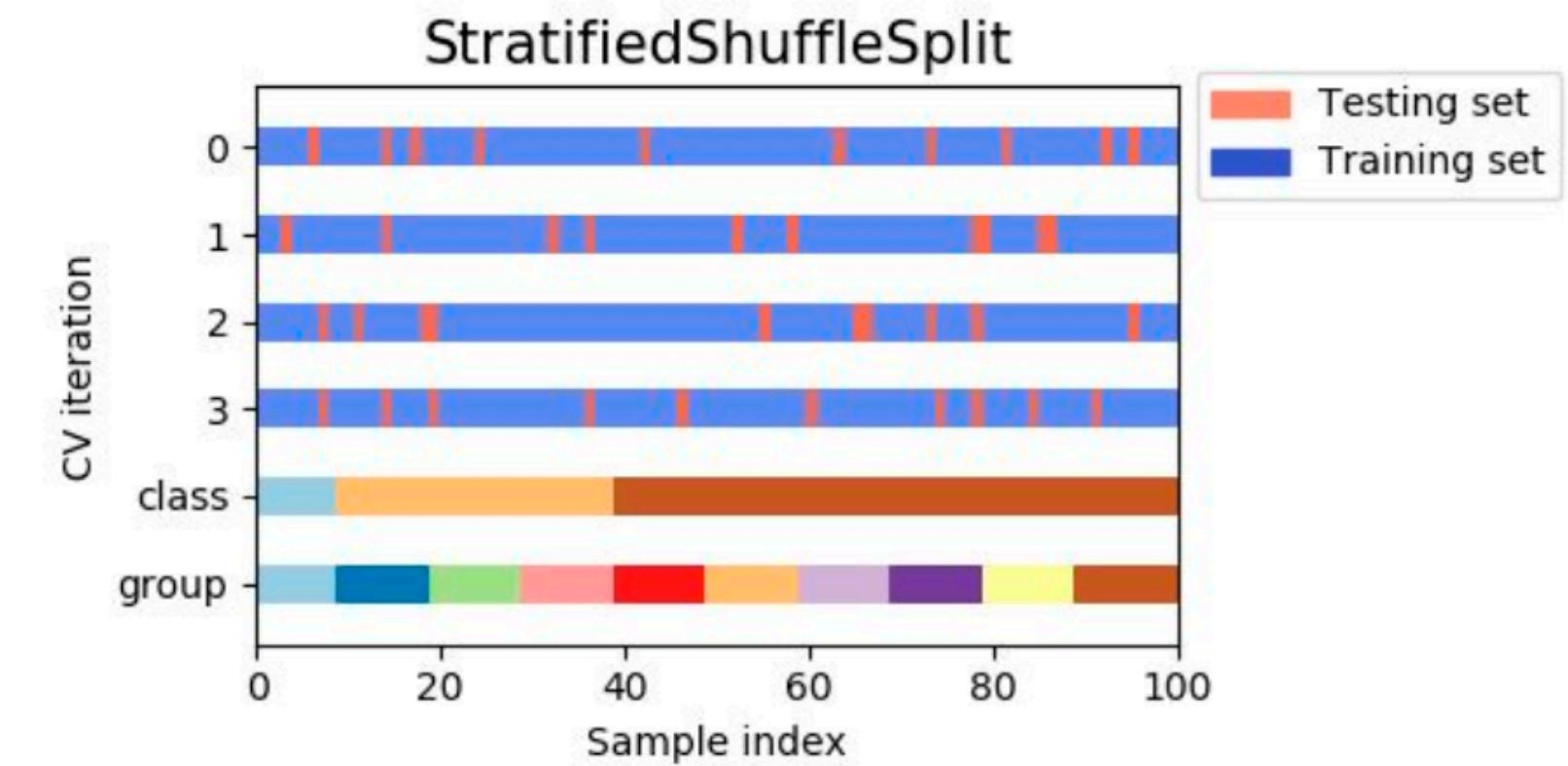
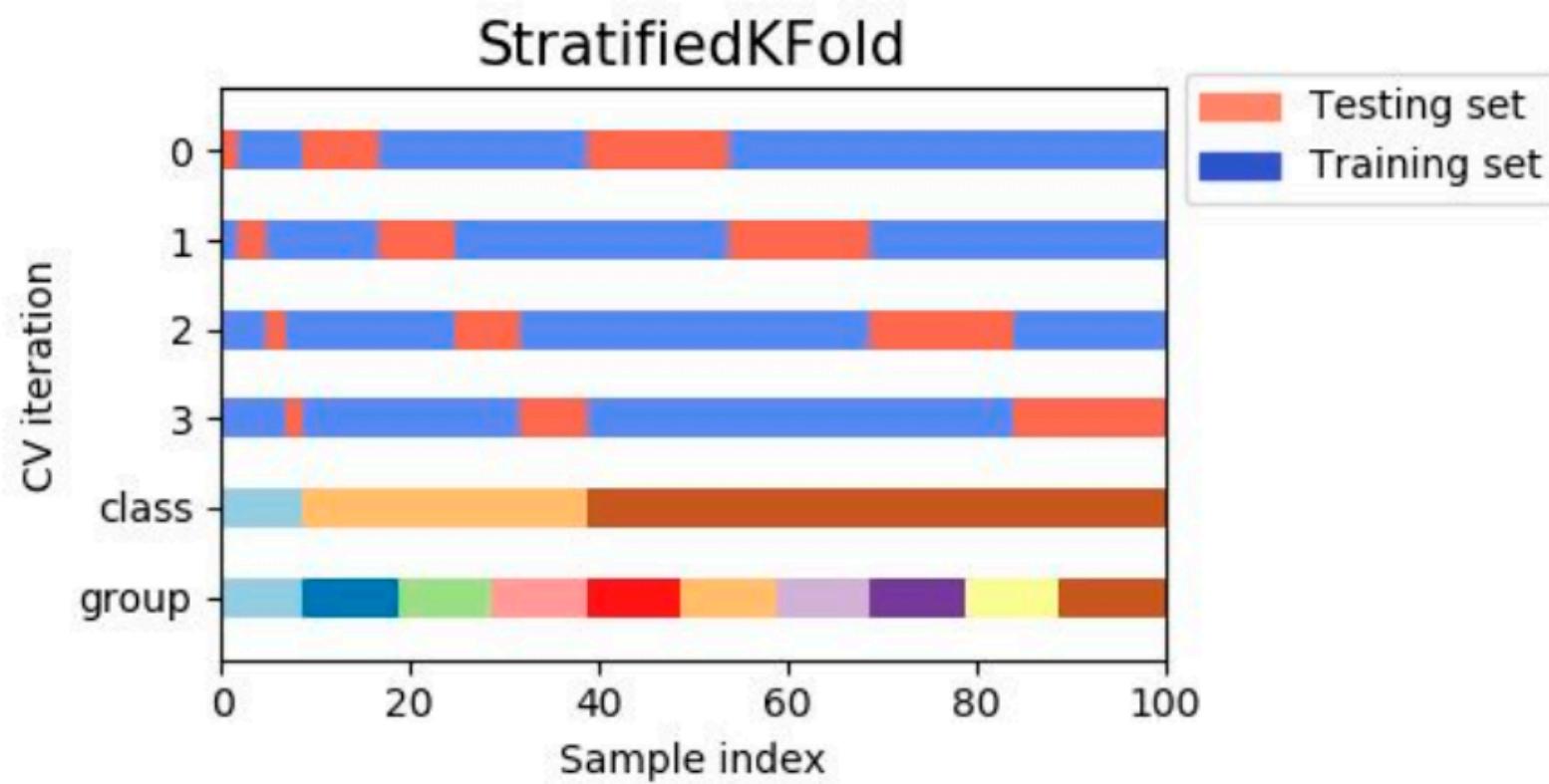
Стратегии валидации



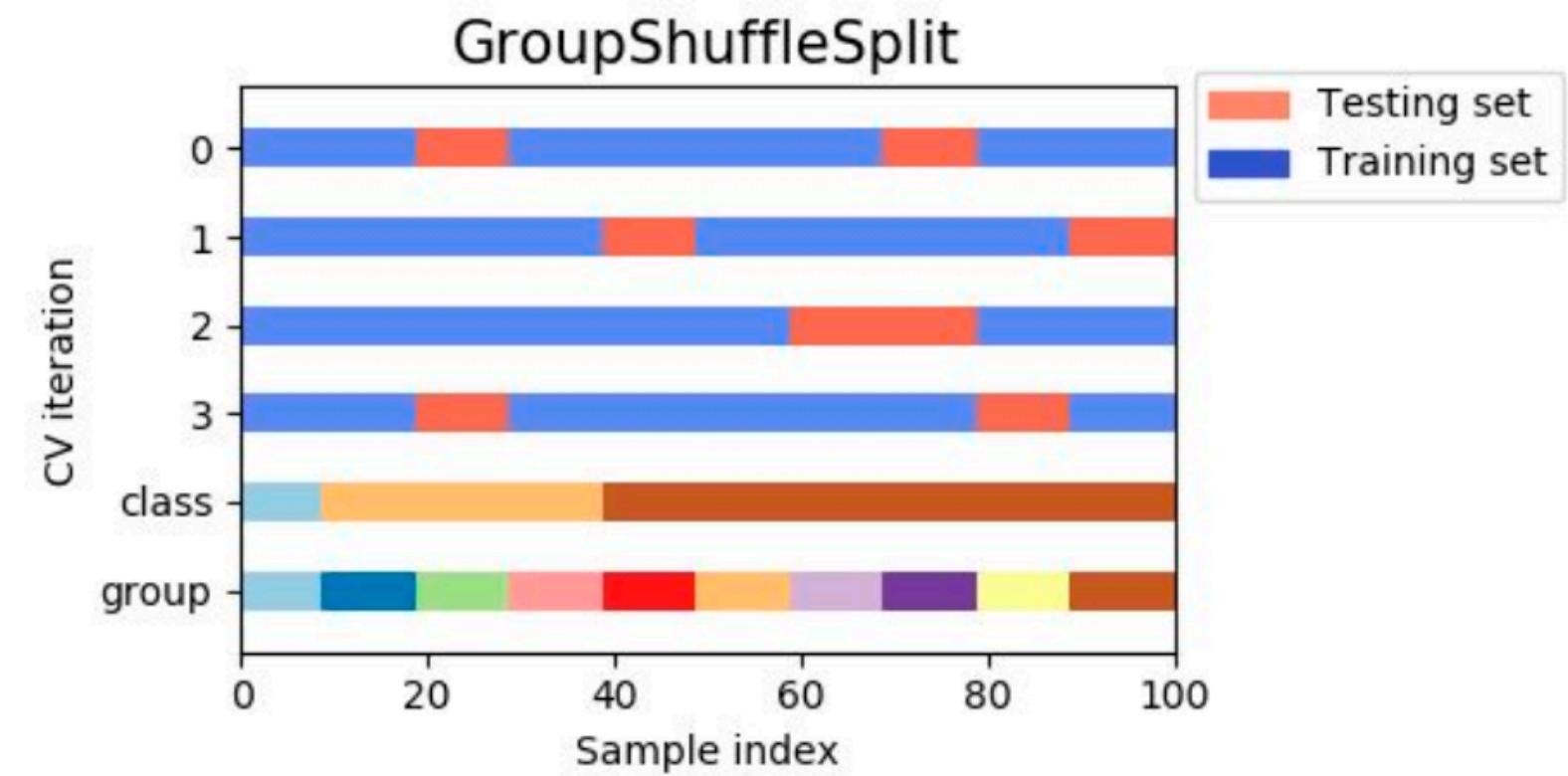
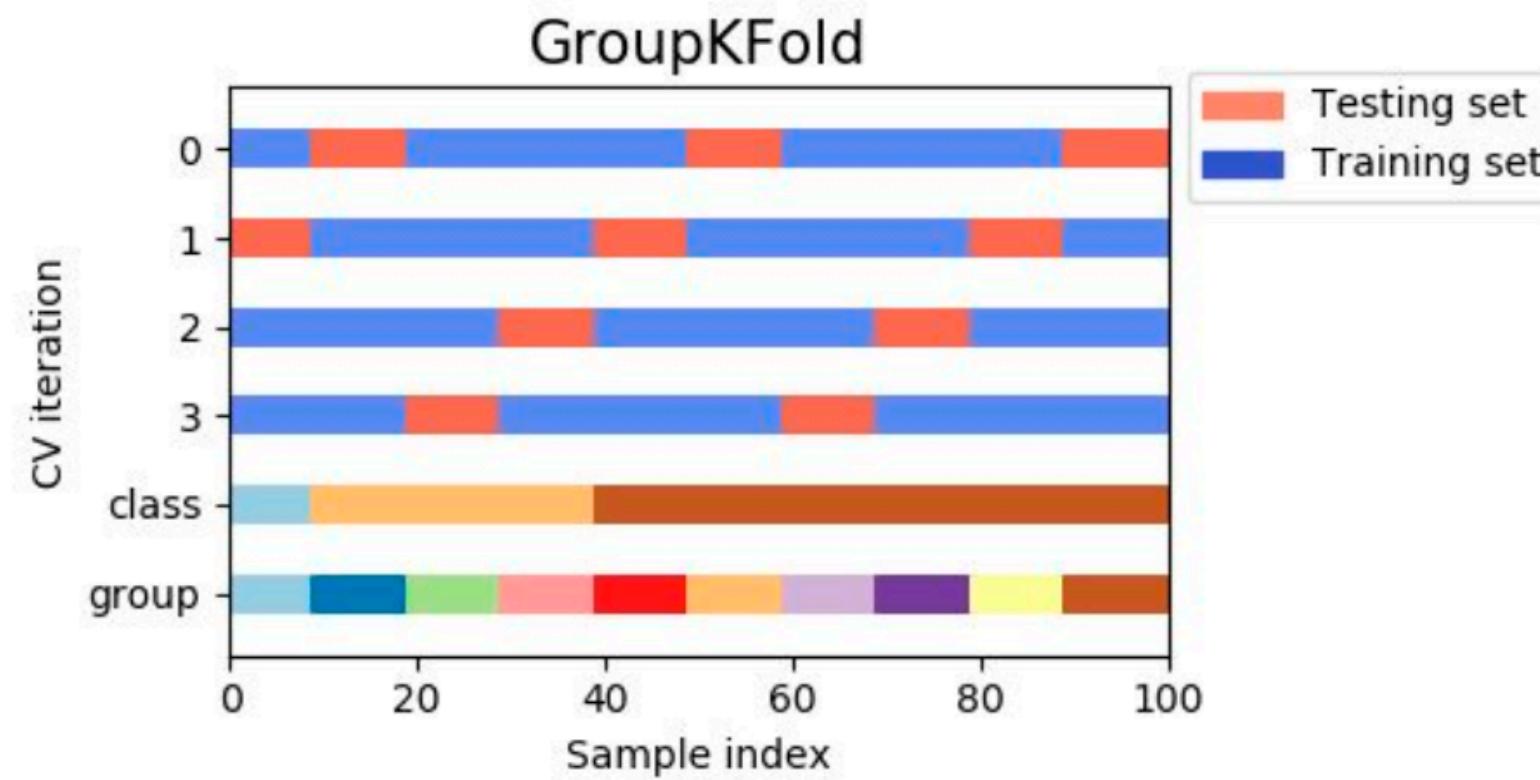
Стратегии валидации



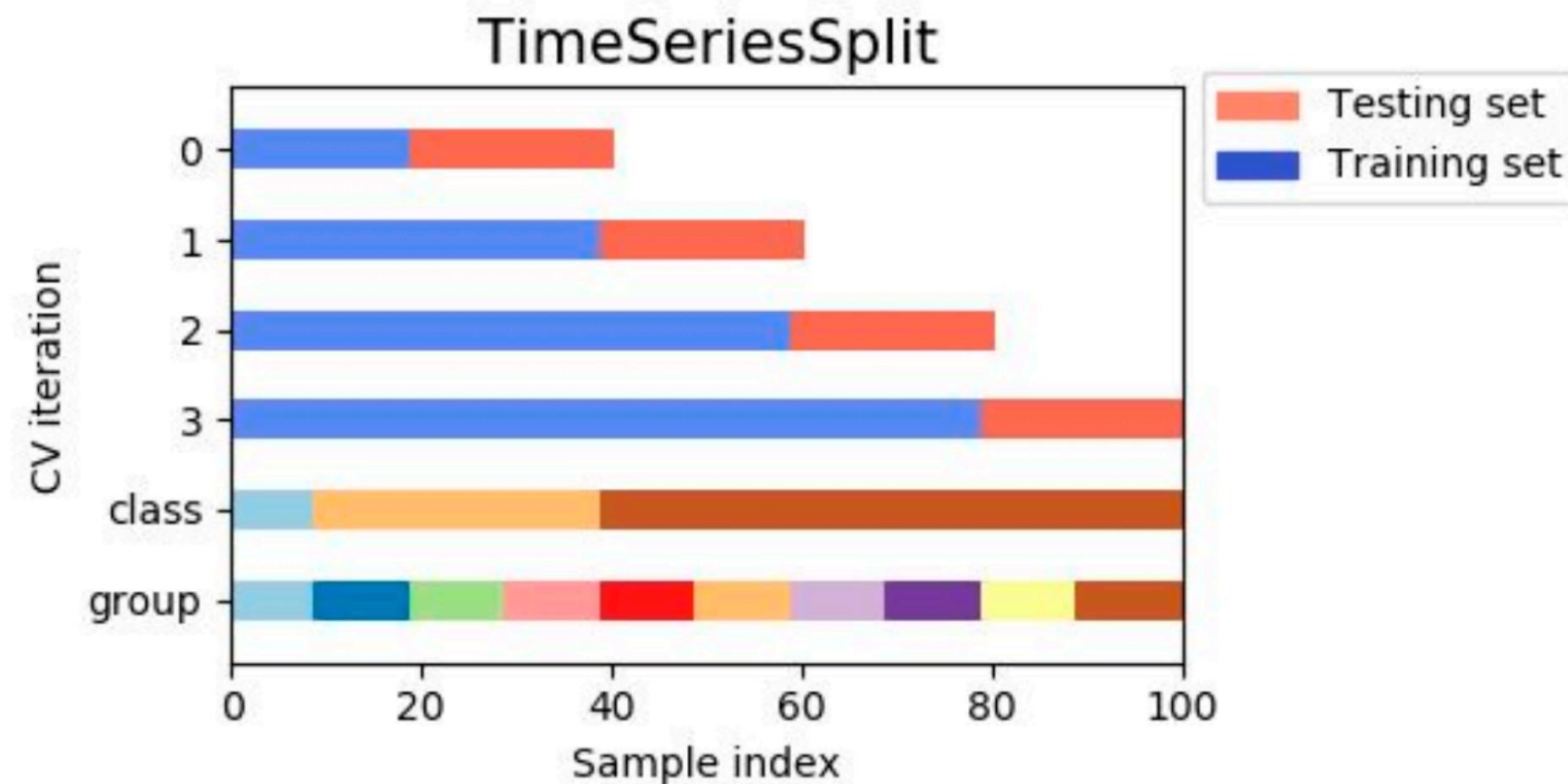
Стратегии валидации



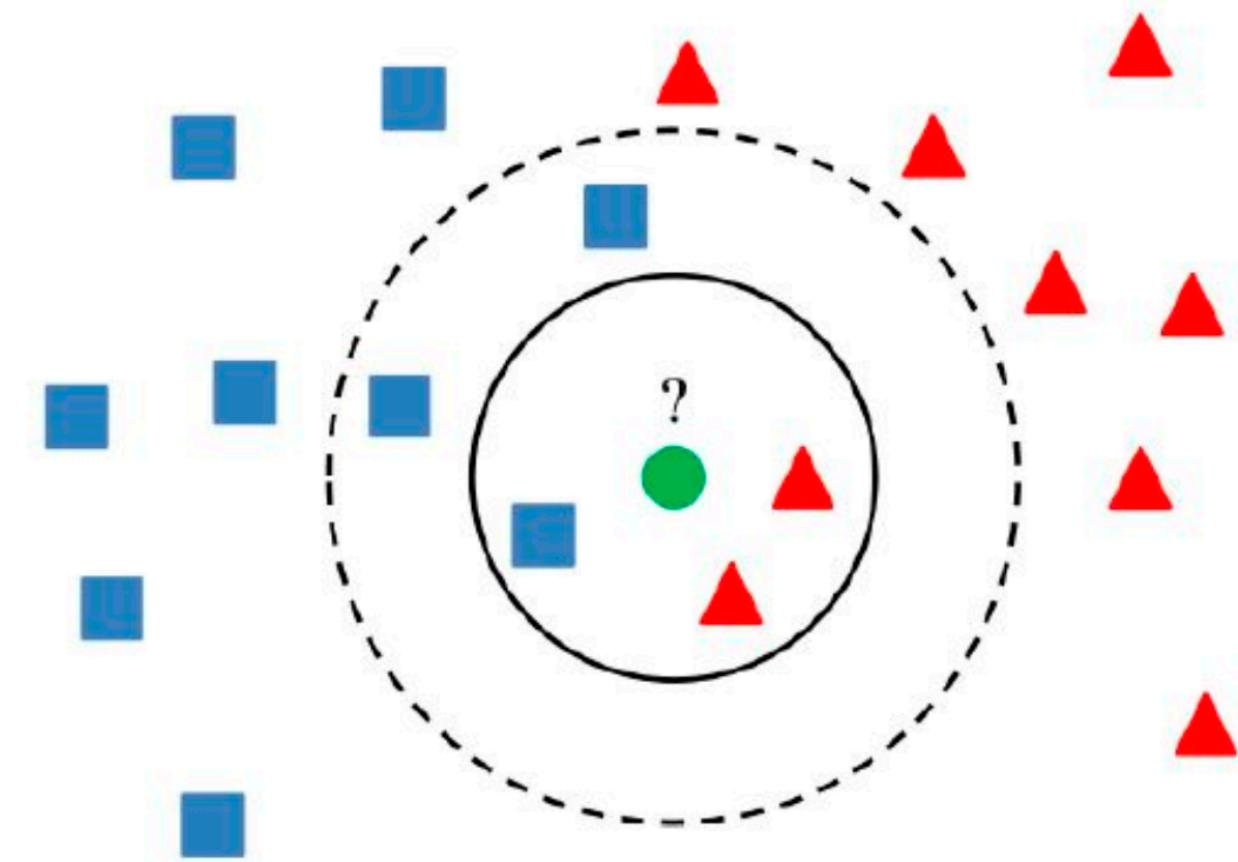
Стратегии валидации



Стратегии валидации



Метод k ближайших соседей (kNN, *k* Nearest Neighbours)



Метод k ближайших соседей (kNN, k Nearest Neighbours)

Выбор метрики

Евклидово расстояние (“euclidean”)

$$\sqrt{\sum (x - y)^2}$$

Расстояние городских кварталов

«манхэттенское расстояние» (“manhattan”)

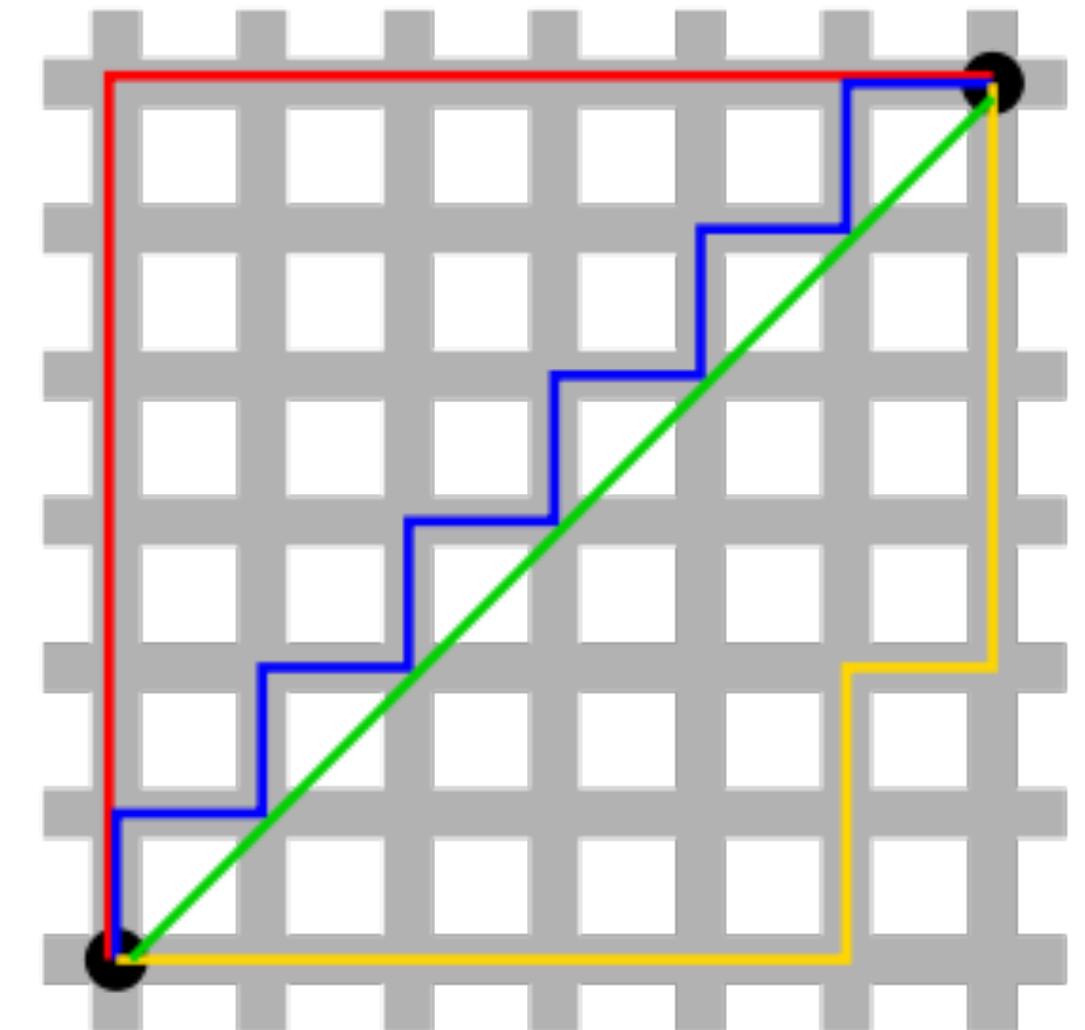
$$\sum |x - y|$$

Расстояние Чебышева “chebyshev”

$$\max(x - y)$$

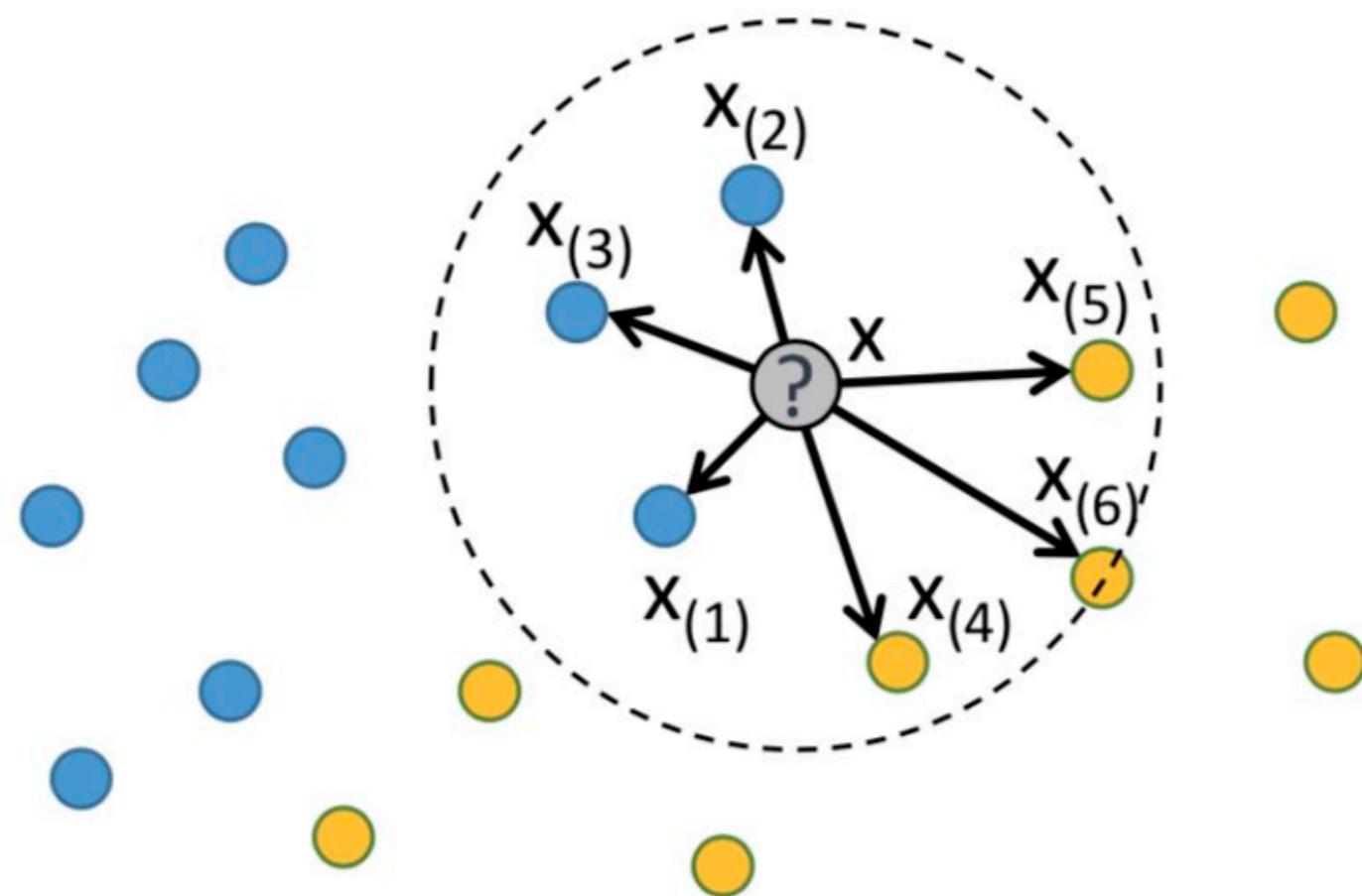
Расстояние Минковского “minkowski”

$$(\sum |x - y|^p)^{\frac{1}{p}}$$



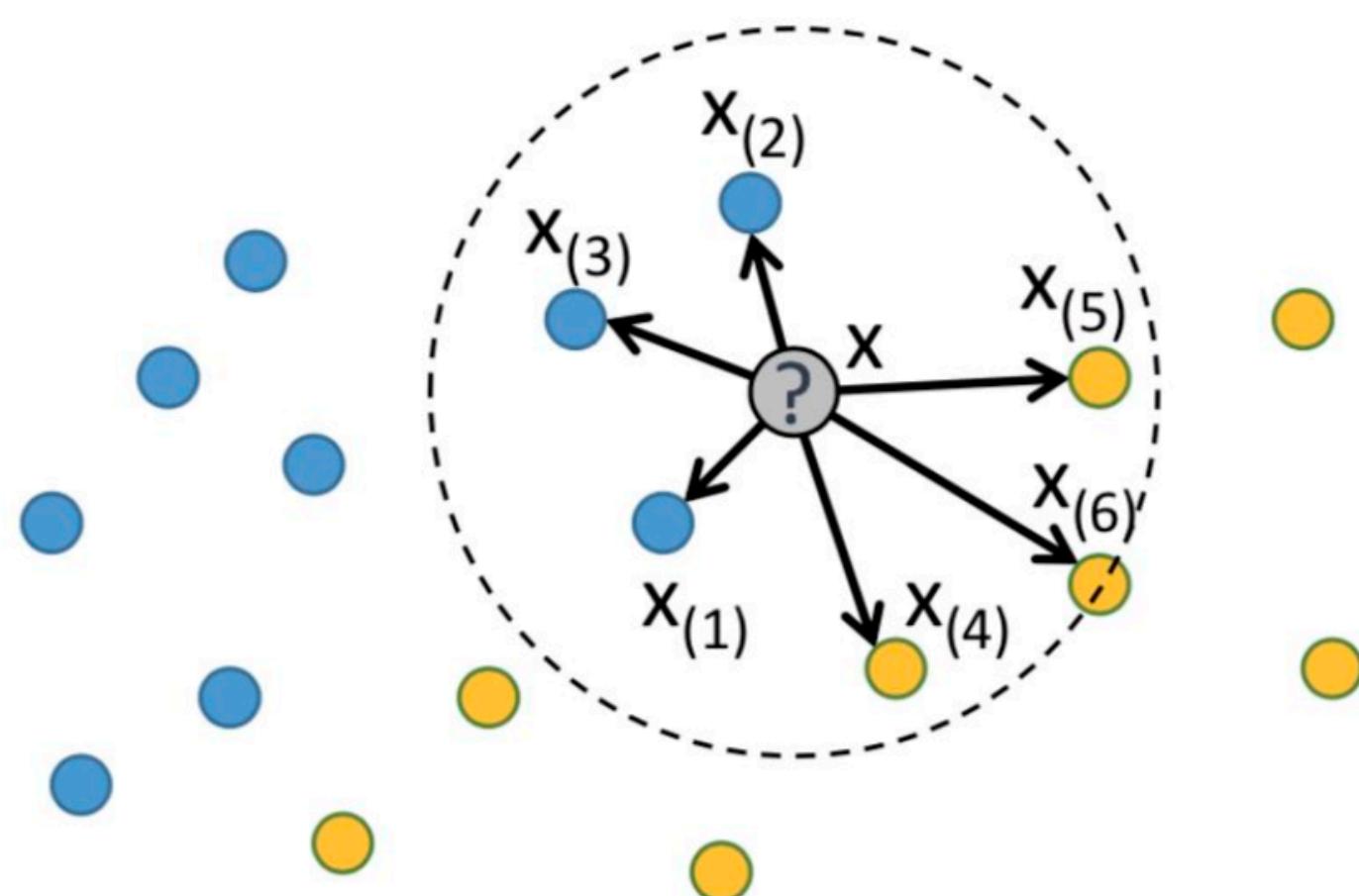
Метод k ближайших соседей (kNN, *k* Nearest Neighbours)

$k = 6$



Метод k ближайших соседей (kNN, *k* Nearest Neighbours)

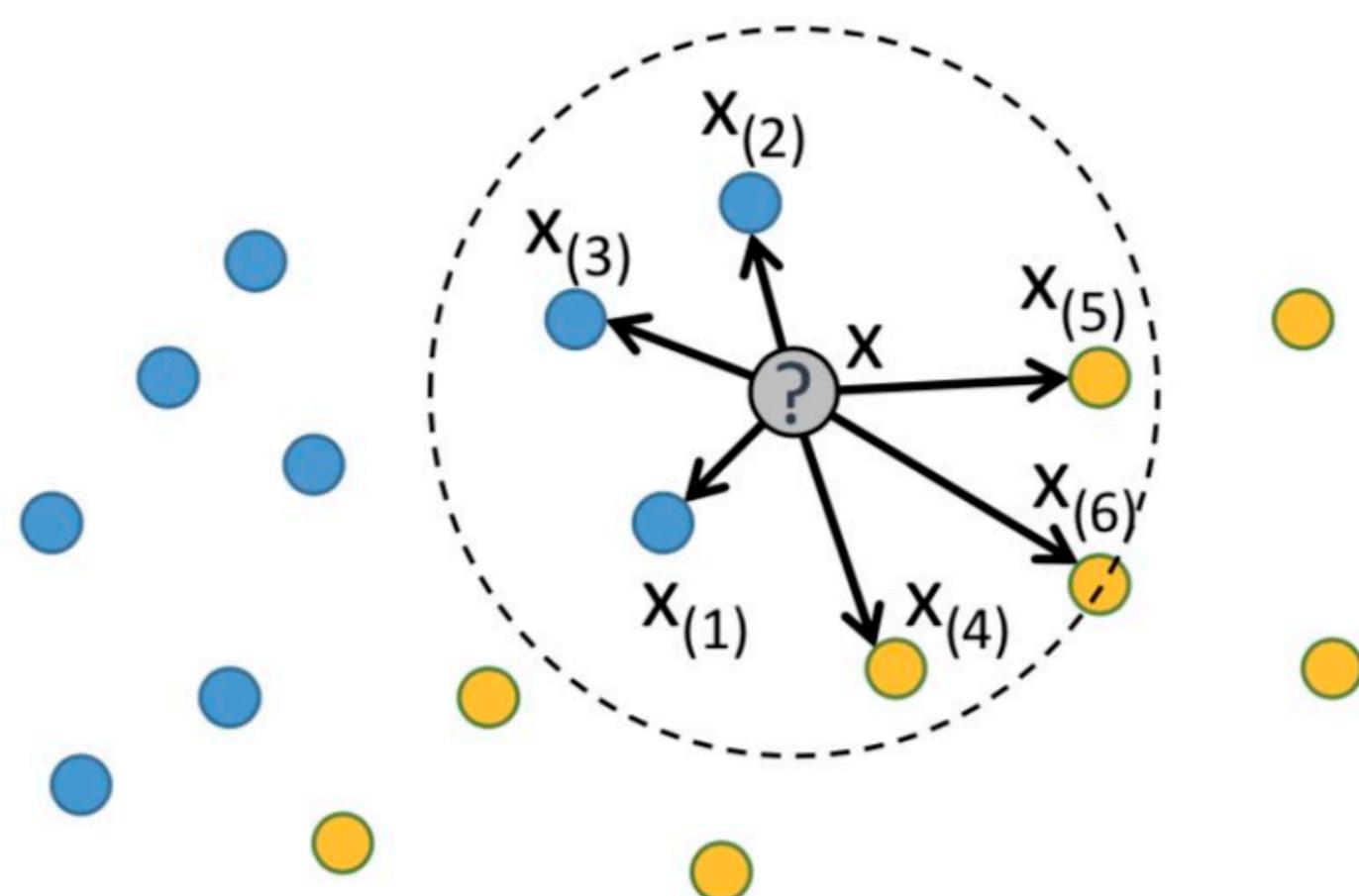
$k = 6$



Веса могут быть выбраны в
соответствии с порядком
объектов или
В соответствии с расстоянием до
объектов

Метод k ближайших соседей (kNN, *k* Nearest Neighbours)

$k = 6$



Веса могут быть выбраны в
соответствии с порядком
объектов или
В соответствии с расстоянием до
объектов

$$z_{\text{blue}} = \frac{w(x_{(1)}) + w(x_{(2)}) + w(x_{(3)})}{w(x_{(1)}) + w(x_{(2)}) + w(x_{(3)}) + w(x_{(4)}) + w(x_{(5)}) + w(x_{(6)})}$$

$$z_{\text{yellow}} = \frac{w(x_{(4)}) + w(x_{(5)}) + w(x_{(6)})}{w(x_{(1)}) + w(x_{(2)}) + w(x_{(3)}) + w(x_{(4)}) + w(x_{(5)}) + w(x_{(6)})}$$

Метод k ближайших соседей (kNN, *k Nearest Neighbours*)

Гипотеза компактности: если мера сходства объектов введена достаточно удачно, то схожие объекты гораздо чаще лежат в одном классе, чем в разных.

1. Вычислить расстояние до каждого из объектов обучающей выборки.
2. Отобрать k объектов обучающей выборки, расстояние до которых минимально.
3. Класс классифицируемого объекта – это класс, наиболее часто встречающийся среди k ближайших соседей.



[`sklearn.neighbors.KNeighborsRegressor`](#)

*(n_neighbors=5, weights='uniform', algorithm='auto',
leaf_size=30, p=2, metric='minkowski',
metric_params=None, n_jobs=None, **kwargs)*

[`sklearn.neighbors.KNeighborsClassifier`](#)

*(n_neighbors=5, weights='uniform', algorithm='auto',
leaf_size=30, p=2, metric='minkowski',
metric_params=None, n_jobs=None, **kwargs)*

Метод k ближайших соседей (kNN, *k Nearest Neighbours*)

Нормирование признаков

$$z = \frac{x - \min(x)}{\max(x) - \min(x)}$$

$$z \in [0, 1]$$

μ - Математическое ожидание (среднее)

σ - Стандартное отклонение $\sigma = \sqrt{D}$

D - Дисперсия $D = \mu(x^2) - \mu(x)^2$

Стандартизация признаков

$$z = \frac{x - \mu(x)}{\mu(x) - \sigma}, \text{где}$$

Рассмотрим описанное выше на практике