











План

- Постановка задачи МЛ
- Обзор направлений
- Типы данных
- Обучение с учителем
- Метрики качества

Постановка задачи МЛ: оцените стоимость ноутбука

		Кол-во ядер	RAM (Гб)	Объем жесткого диска (ГБ)	Диагональ/ разрешение	Работа от аккумулятора	Цена (руб.)
1		2	4	500 (HDD)	15"/1920x1080 пикс.	до 5 часов	31 490
2		4	8	256 (SSD)	14"/1920x1080 пикс.	до 12 часов	60 990
3		4	16	1000 (HDD)	17"/1920x1080 пикс.	до 3 часов	65 990
4		8	16	1000 (HDD) + 256 (SSD)	17"/1920x1080 пикс.	до 11 часов	109 990
5		4	16	1000 (HDD)+ 128 (SSD)	17"/1920x1080 пикс.	до 6 часов	?

Постановка задачи МЛ: оцените стоимость ноутбука

		Кол-во ядер	RAM (Гб)	Объем жесткого диска (ГБ)	Диагональ/ разрешение	Работа от аккумулятора	Цена (руб.)
1		2	4	500 (HDD)	15"/1920x1080 пикс.	до 5 часов	31 490
2		4	8	256 (SSD)	14"/1920x1080 пикс.	до 12 часов	60 990
3		4	16	1000 (HDD)	17"/1920x1080 пикс.	до 3 часов	65 990
4		8	16	1000 (HDD) + 256 (SSD)	17"/1920x1080 пикс.	до 11 часов	109 990
5		4	16	1000 (HDD)+ 128 (SSD)	17"/1920x1080 пикс.	до 6 часов	86990

Постановка задачи машинного обучения

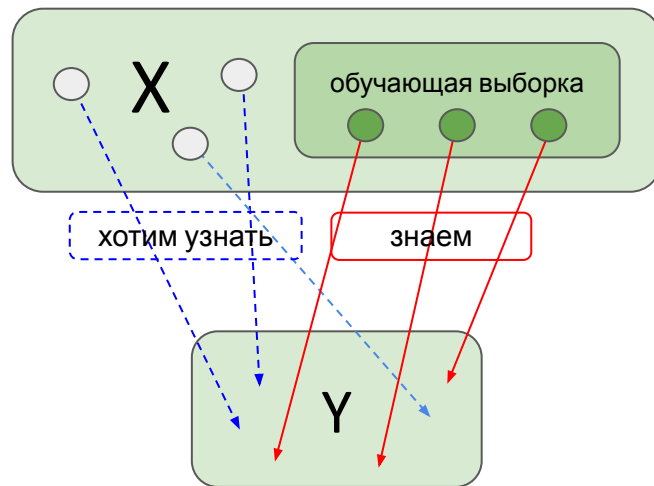
X — множество *объектов*

Y — множество *ответов* (например, два класса или произвольные числа)

$y: X \rightarrow Y$ — неизвестная закономерность

Дано: обучающая выборка, $\{x_1, x_2, \dots, x_n\}$ — подмножество множества X

Цель: подобрать *алгоритм*, приближающий функцию $y(x)$.



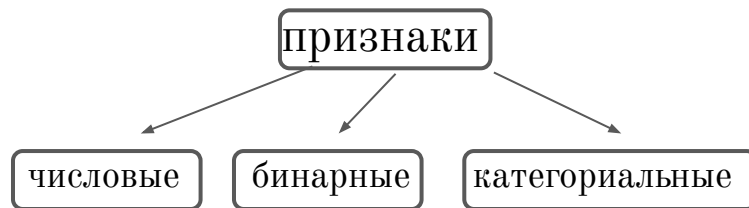
Как задаются объекты. Признаковое описание

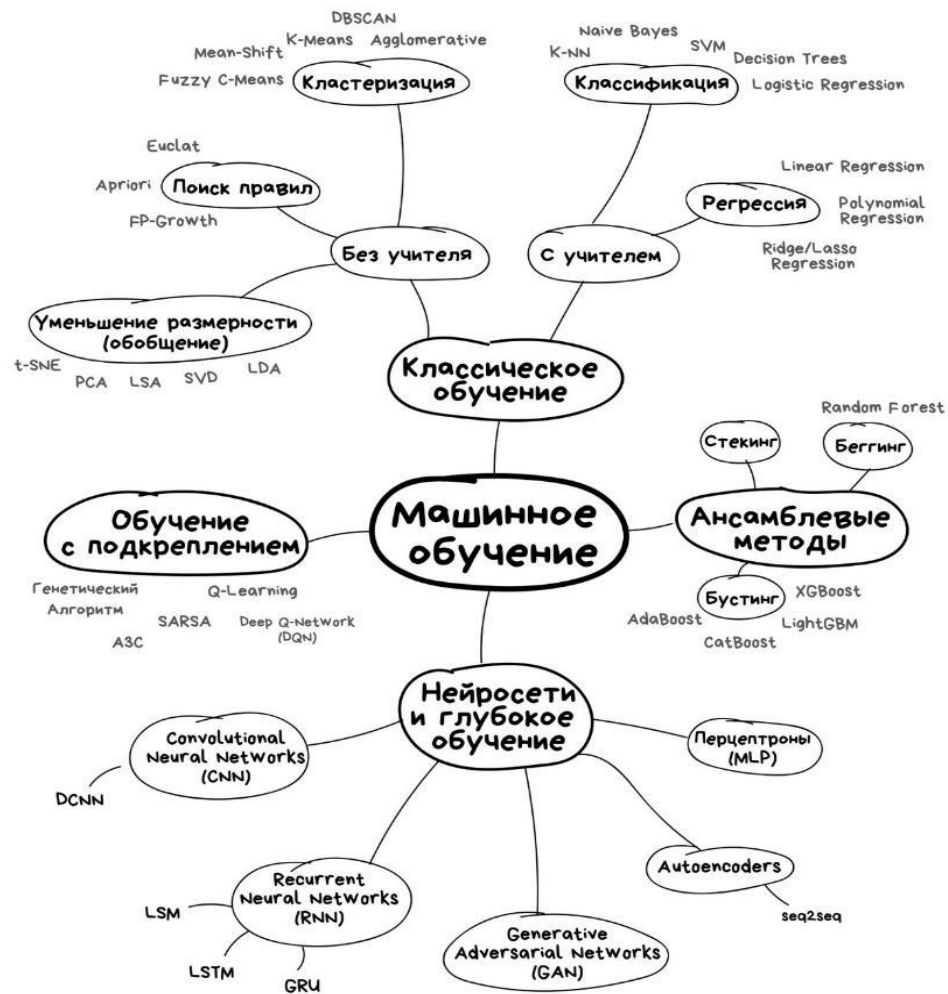
Объект x задаётся *признаковым описанием*

f_1, f_2, \dots, f_k — признаки (features) объекта x

$$\begin{bmatrix} f_1(x_1), & f_2(x_1), & \dots, & f_k(x_1) \\ f_1(x_2), & f_2(x_2), & \dots, & f_k(x_2) \\ & & \dots & \\ f_1(x_n), & f_2(x_n), & \dots, & f_k(x_n) \end{bmatrix}$$

— матрица “объекты-признаки”
объект, пригодный для применения
алгоритмов машинного обучения





Не все так просто: типы обучения

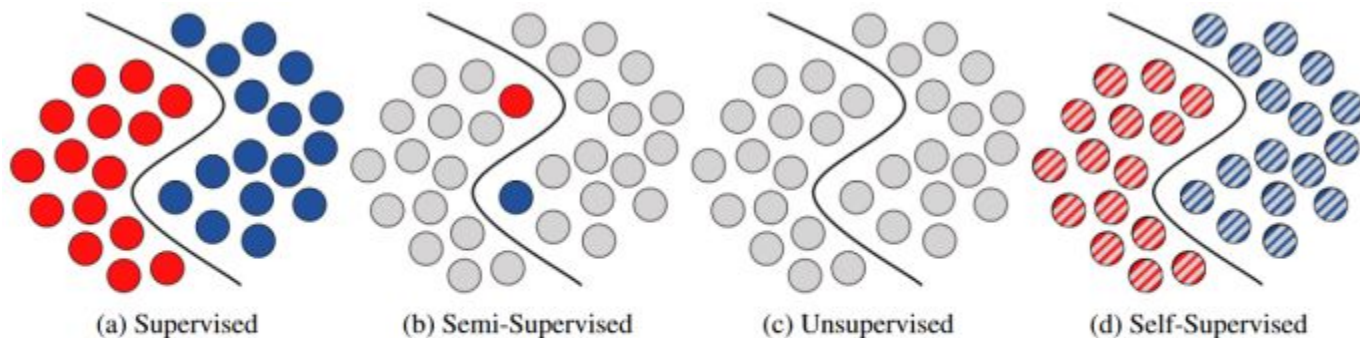


Figure 2: Illustrations of the four presented deep learning strategies - The red and dark blue circles represent labeled data points of different classes. The light grey circles represent unlabeled data points. The black lines define the underlying decision boundary between the classes. The striped circles represent datapoints which ignore and use the label information at different stages of the training process.

Обучение по размеченным данным (Supervised Learning / SL)

- Обучающая выборка состоит из пар (x, y) , где x – описание объекта, y – его метка
- Необходимо обучить модель $y=f(x)$, которая по описаниям получает метки. Часто такое обучение называют «обучением с учителем»

Обучение с частично размеченными данными (Semi-Supervised Learning / SSL)

- Обучающая выборка состоит из данных с метками и без меток (последних, как правило, существенно больше)
- Необходимо также обучить модель $y=f(x)$, но здесь может помочь информация о том, как объекты располагаются в пространстве описаний

Обучение по неразмеченным данным (Unsupervised Learning / UL)

- Даны только объекты (без меток), необходимо эффективно описать, как они располагаются в пространстве описаний.
- Типичные задачи обучения по неразмеченным данным – кластеризация, понижение размерности, детектирование аномалий, оценка плотности и т.п.

Самообучения (Self-Supervised Learning).

- Необходимо сформировать для каждого объекта псевдо-метку (pseudo label) и решить полученную SL-задачу, но нас интересует не столько качество решения придуманной нами задачи (её называют pretext task), сколько представление (representation) объектов, которое будет выучено в ходе её решения.
- Это представление можно в дальнейшем использовать уже при решении любой задачи с метками (SL), которую называют последующей задачей (downstream task).

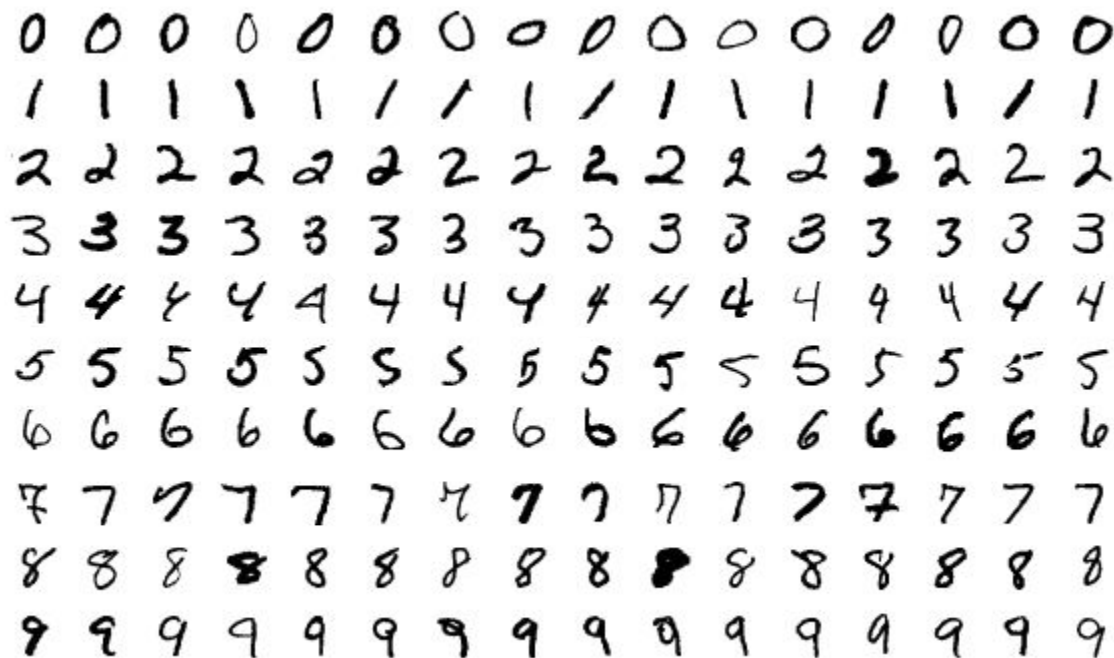
Причина в небольшой объёме размеченных данных (в этом случае возникает соблазн использовать неразмеченные данные, возможно из другого домена). В отличие от обучения с частично размеченными данными в самообучении используются совершенно произвольные неразмеченные данные (не имеющие отношения к решаемой задаче).

Данные

Изображения

MNIST Dataset

- Изображения цифр, написанных от руки
- ~50к изображений
- Можно научить модель распознавать цифру



Табличные данные

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket
1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17599
3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 310128
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803
5	0	3	Allen, Mr. William Henry	male	35	0	0	373450
6	0	3	Moran, Mr. James	male		0	0	330877
7	0	1	McCarthy, Mr. Timothy J	male	54	0	0	17463
8	0	3	Palsson, Master. Gosta Leonard	male	2	3	1	349909

- Таблица как в Excel
- Как часто бывает: десятки столбцов, тысячи строк
- Один из столбцов - целевая переменная
- С данными такого вида мы будем работать на протяжении всего курса

Признаковое описание

Для того, чтобы работать с данными, нужно представить их в виде, пригодном для моделей ML

- Строка в таблице называется **объектом**
- Столбец в таблице называется **признаком**
- Признаки могут быть 3-х типов:
 - Числовые
 - Категориальные
 - Бинарные
- Столбец, который нужно предсказать, называется **целевой переменной**

X y^* features

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket
1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17599
3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 310128
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803
5	0	3	Allen, Mr. William Henry	male	35	0	0	373450
6	0	3	Moran, Mr. James	male		0	0	330877
7	0	1	McCarthy, Mr. Timothy J	male	54	0	0	17463
8	0	3	Palsson, Master. Gosta Leonard	male	2	3	1	349909

Признаковое описание

Все признаки представляются в виде чисел:

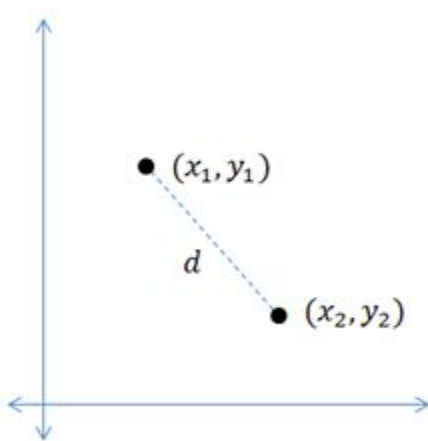
- Числовые признаки - это уже числа
- Бинарные признаки - как 0 и 1
- Категориальные признаки:
 - Как число от 0 до N, где N - число категорий
 - Как N-мерный вектор {0, 0, 1, 0, 0, 0}. Т.н. **one-hot vector**

Для каждого объекта набор его признаков собирается в один вектор

Вектор

- Вектор - это упорядоченный набор чисел
- Вектор - это координаты точки в пространстве

Для двух точек можно рассчитать расстояние между ними



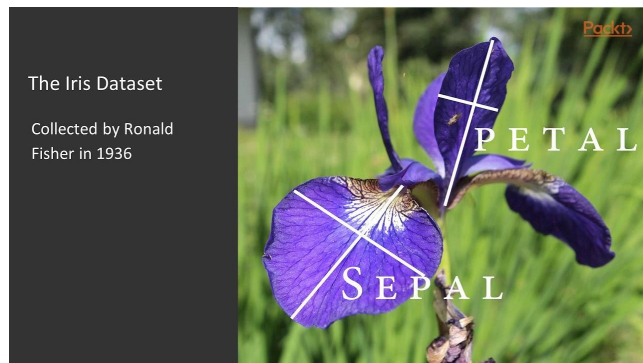
$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Матрица

- Матрица - это упорядоченный набор векторов одного размера
- Набор векторов - это набор точек в пространстве

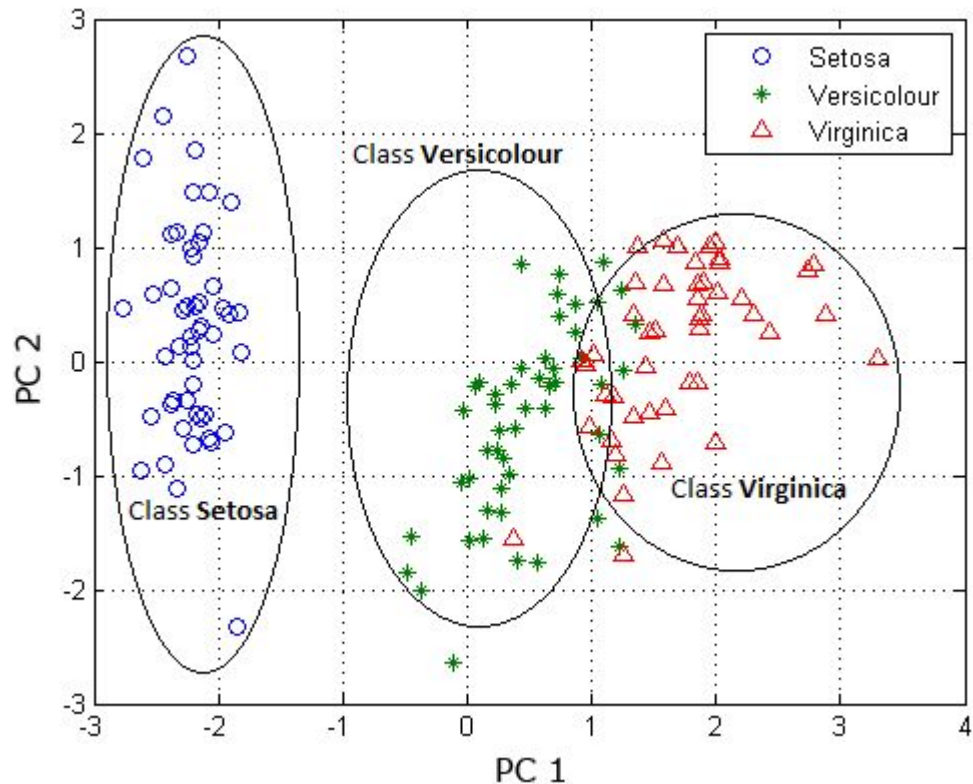
Датасет с подготовленными признаками - это матрица

Визуализация данных



Датасет - это матрица

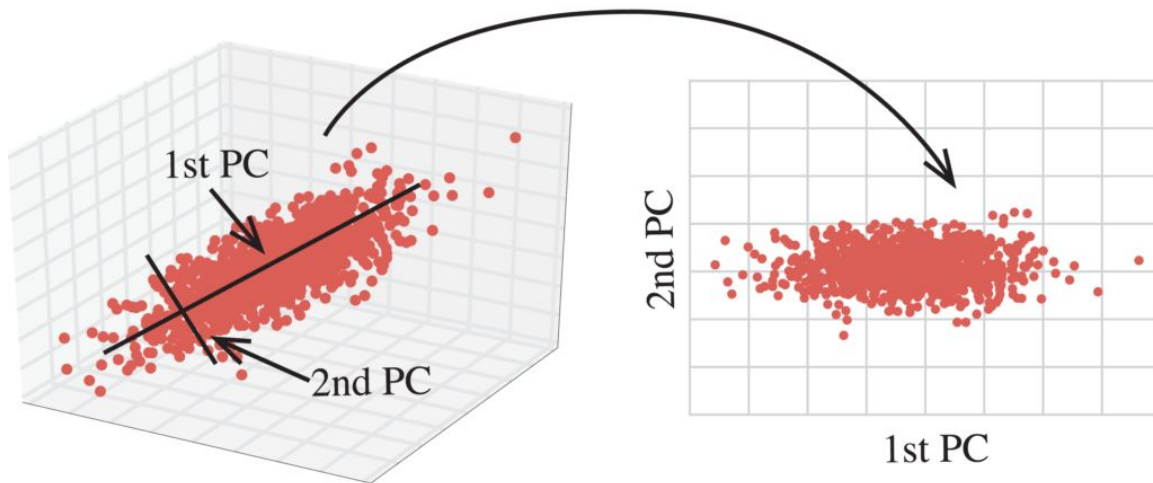
petal length	petal width	target
1.4	0.2	Iris-setosa
1.4	0.2	Iris-setosa
1.3	0.2	Iris-setosa
1.5	0.2	Iris-setosa
1.4	0.2	Iris-setosa



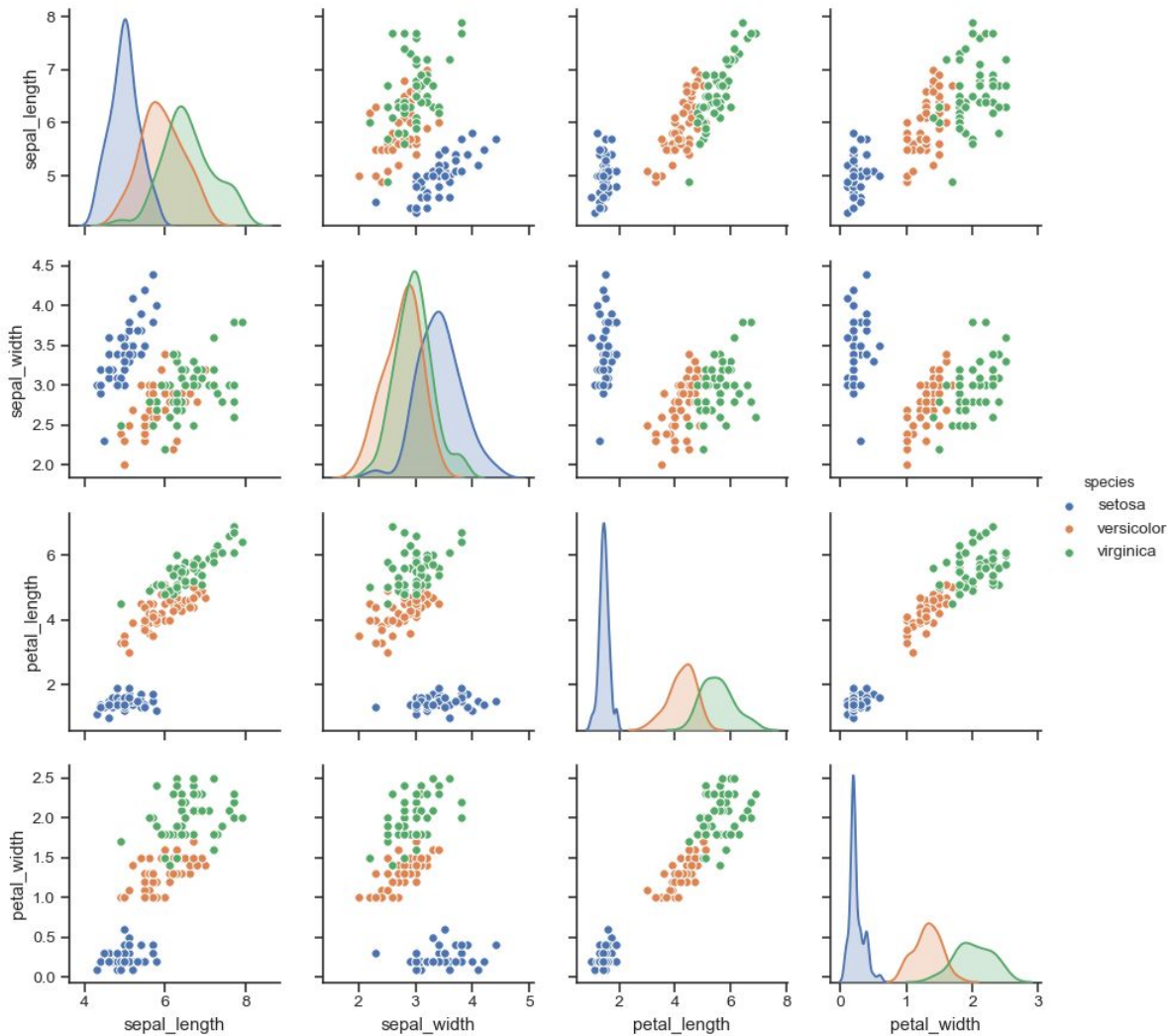
Вообще-то, в этом датасете признаков не 2, а 4. Как на них смотреть?

Большие размерности

- Двумерный набор точек можно нарисовать на плоскости
- Трёхмерный набор можно спроецировать на плоскость
- Размерности векторов могут быть порядков 100~100 000
- Их всё равно можно спроецировать!



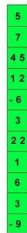
Визуализация реальных датасетов



Тензор

- Упорядоченный набор точек - это вектор
- Упорядоченный набор векторов одного размера - это матрица
- Упорядоченный набор матриц одного размера - это **тензор**

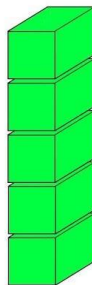
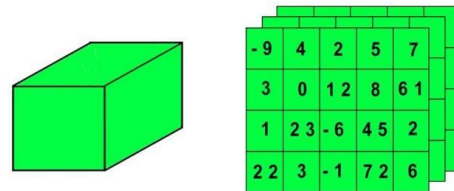
1D TENSOR /
VECTOR



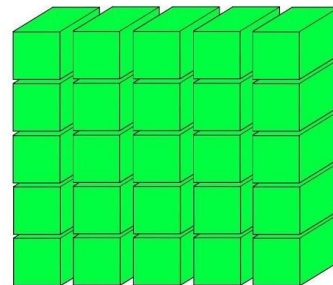
2D TENSOR /
MATRIX

- 9	4	2	5	7
3	0	1 2	8	6 1
1	2 3	- 6	4 5	2
2 2	3	- 1	7 2	6

3D TENSOR /
CUBE



4D TENSOR
VECTOR OF CUBES



5D TENSOR
MATRIX OF CUBES

Tensor shape

У тензора k-го ранга есть k индексов

Форма тензора k-го ранга - это набор k чисел, каждое из которых означает: какое кол-во значений может пробегать данный индекс. Т.е. протяженность тензора в разных направлениях.

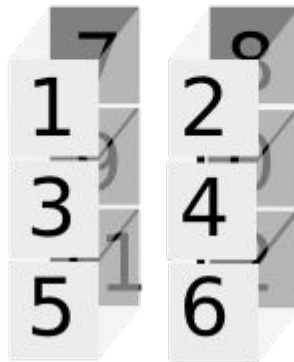
Форма тензора справа равна [2, 3, 2]

Форма цветного FullHD изображения: [1920, 1080, 3]

*Слово “форма” по-русски не применяется для тензоров. Это дословный перевод термина “shape”

This is how you represent a tensor in code

```
[ [[1,2], [3,4], [5,6]], [[7,8], [9,10], [11,12]] ]
```

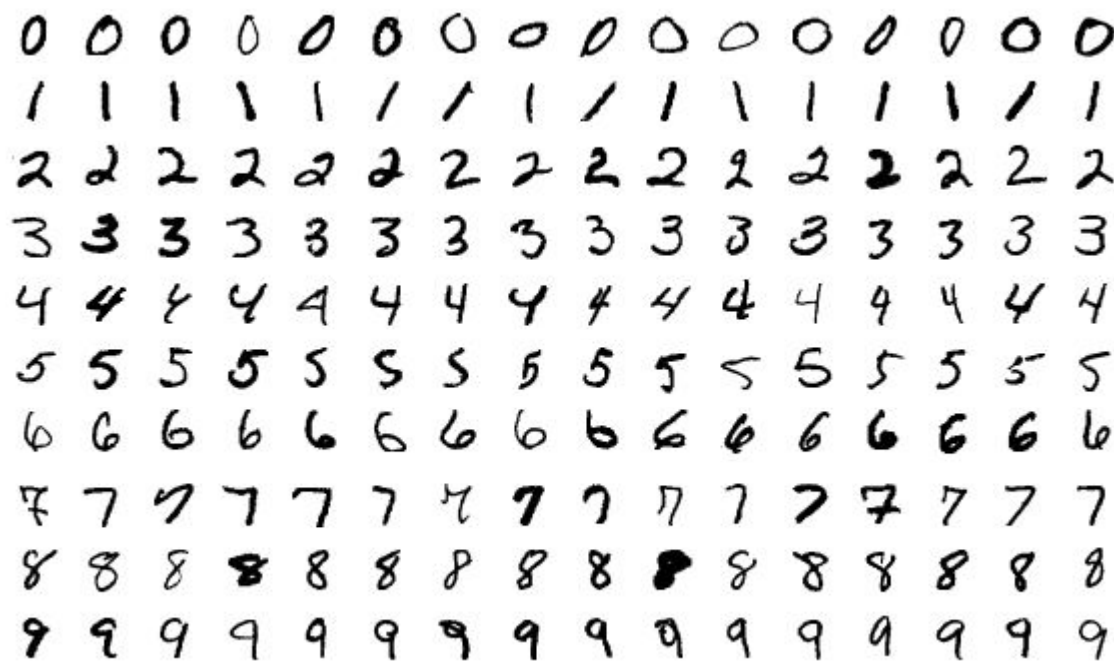


And this is how we like to visualize tensors

MNIST

Вопрос: каким тензором является датасет?

Вопрос: что является объектом, признаком, и целевой переменной в задаче MNIST? Какого они типа?



Модель

Задача обучения с учителем

- Между объектом и целевой переменной существует **реальная зависимость**
- У нас есть только N сэмплов этой зависимости - **обучающая выборка**
- Задача - научиться **предсказывать** целевую переменную для новых точек
- Для этого строится **модель**

Модель - это функция, которой можно аппроксимировать реальную зависимость, имея конечное число примеров.

Классификация и регрессия

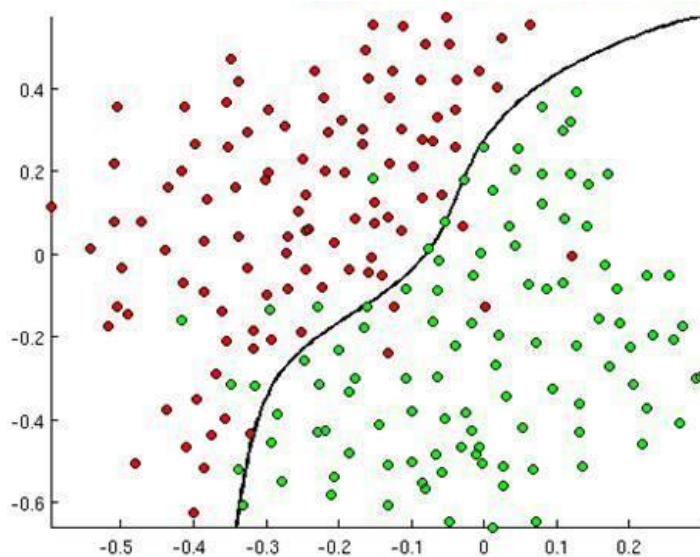
Целевая переменная, как и признаки, может быть трех типов:

- Числовая
- Бинарная
- Категориальная

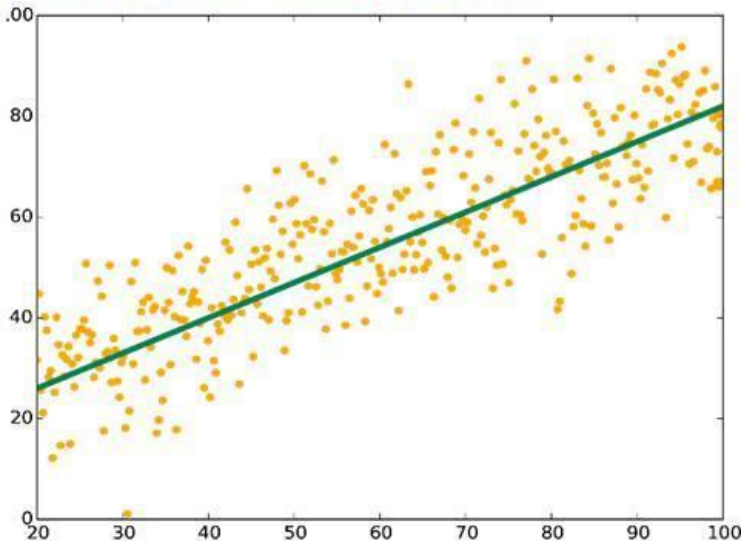
Предсказание числового значения называется **регрессия**

Предсказание одного из нескольких классов называется **классификация**

Классификация и регрессия

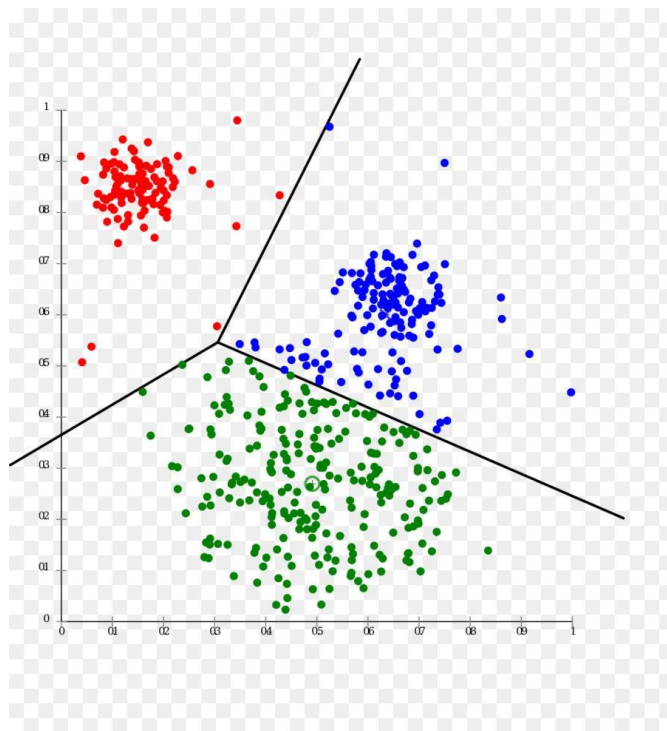


Classification



Regression

Кластеризация



По типу данных

- Табличные -> вся классика (линейные методы, SVM, наивный Байес, деревья, леса и ансамблевые методы), градиентный бустинг, простейшие полносвязные нейросети
- Временные ряды (сюда же звук) -> классические подходы (хольта-винтерса, арима/сарима, линейные методы, деревья), рекуррентные нейросети (LSTM, GRU)
- Изображения -> сверточные сети
- Текст -> рекуррентные нейросети (LSTM), но обычно архитектуры на основе трансформеров

KNN

K Nearest Neighbors

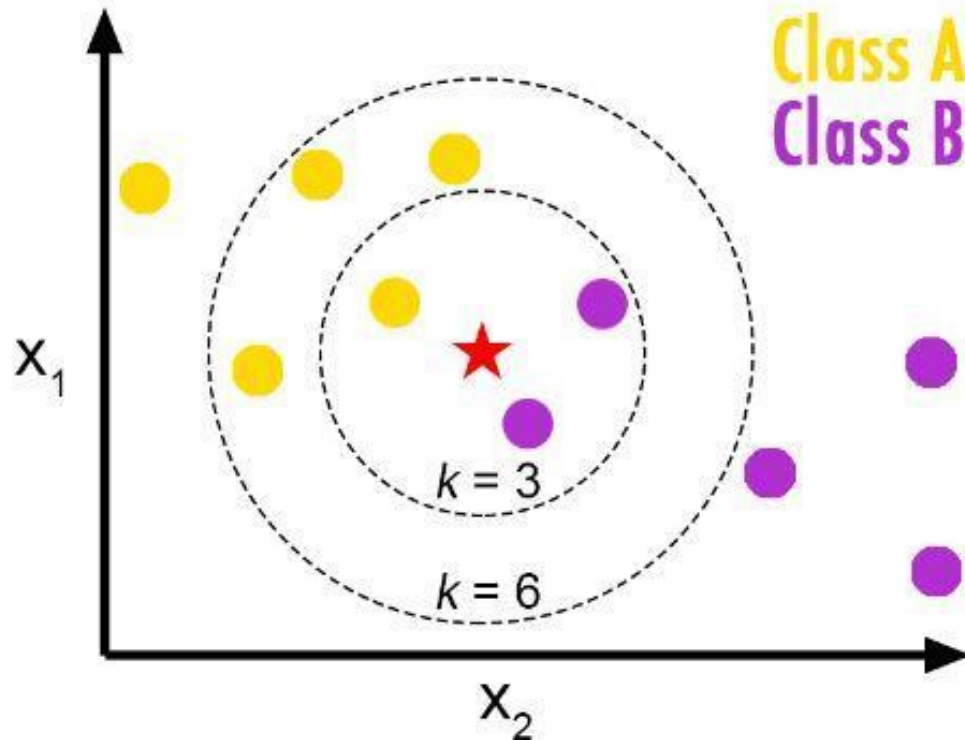
Метод K ближайших соседей

- На вход подается вектор - признаковое описание какого-то объекта
- Находится K ближайших к нему векторов, для которых ответ известен
- Ответ для новой точки выбирается с помощью
 - Усреднения в случае регрессии
 - Голосования в случае классификации
- Возможно также усреднение/голосование с весами

KNN классификация

K - внешний параметр. Он подбирается так, чтобы модель работала как можно лучше.

Результат предсказания для некоторых точек может зависеть от K



Метрики

Измерение качества модели

Чтобы понять, насколько адекватно ведет себя модель, нужно каким-то образом численно оценить ее качество.

Метрика - это функция вида: $metric(\mathbf{y}, \hat{\mathbf{y}})$

где \mathbf{y} - это правильное значение целевой переменной (**label**),

а $\hat{\mathbf{y}}$ - значение, предсказанное моделью (**prediction**).

Примеры метрик

Классификации:

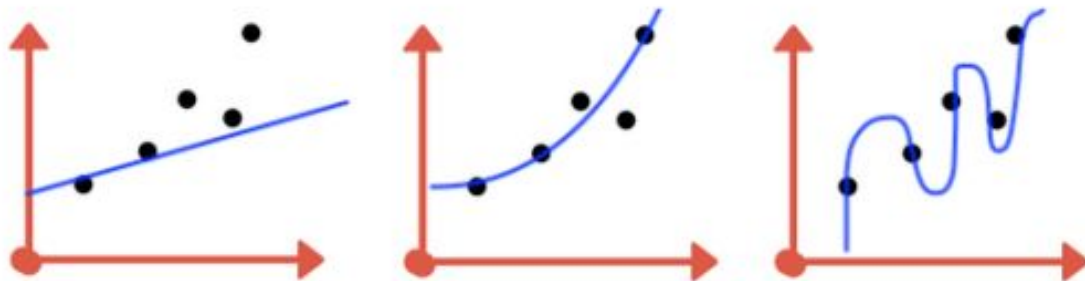
- **accuracy** - процент правильных предсказаний среди всех примеров
- precision - точность
- recall - полнота
- f1 - объединяет полноту и точность
- ROC-AUC - вероятность правильного ранжирования двух случайных примеров

Регрессии:

- MSE - средний квадрат отклонения
- RMSE - стандартное отклонение
- MAE - средний модуль отклонения
- R2 - коэффициент детерминации

Более подробно метрики будут рассмотрены после практического занятия

Несмещенная оценка



Вопрос: какое предсказание лучше по метрикам, а какое на самом деле?

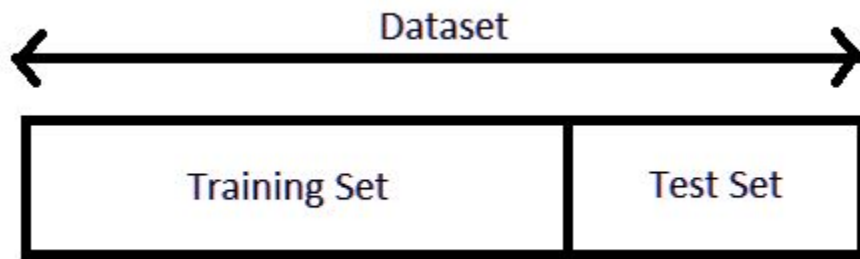
Если тестировать модель на той же выборке, на которой она обучалась, то оценка получится смещенной. В таком случае “самая лучшая” модель - это та, которая просто запомнила все данные.

Хорошая модель должна делать хорошие предсказания на **новых** для себя данных

Отложенная выборка

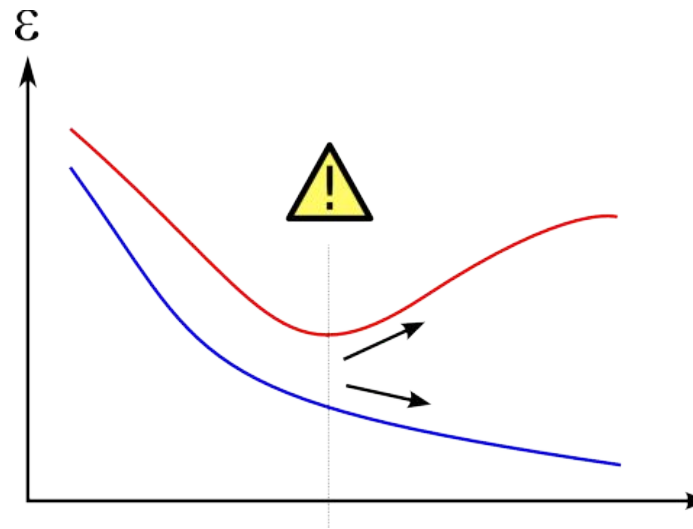
Можно “отложить”, скажем, 20% обучающей выборки для валидации модели. Использовать 80% выборки для обучения и 20% для тестирования.

- Оценка на тестовой выборке будет несмещенной
- Тестовая выборка маленькая - оценка будет иметь погрешность



Переобучение

- Как обнаружить? - Train/Test split
 - Разделить выборку на обучающую и контрольную
 - Следить за качеством на контрольной выборке
- Минусы?
 - Уменьшение размера обучающей выборки может негативно сказаться на качестве
 - Малый размер тестовой выборки может давать сильное смещение оценки.
 - Можно переобучиться под **тестовую выборку**



Кросс-валидация

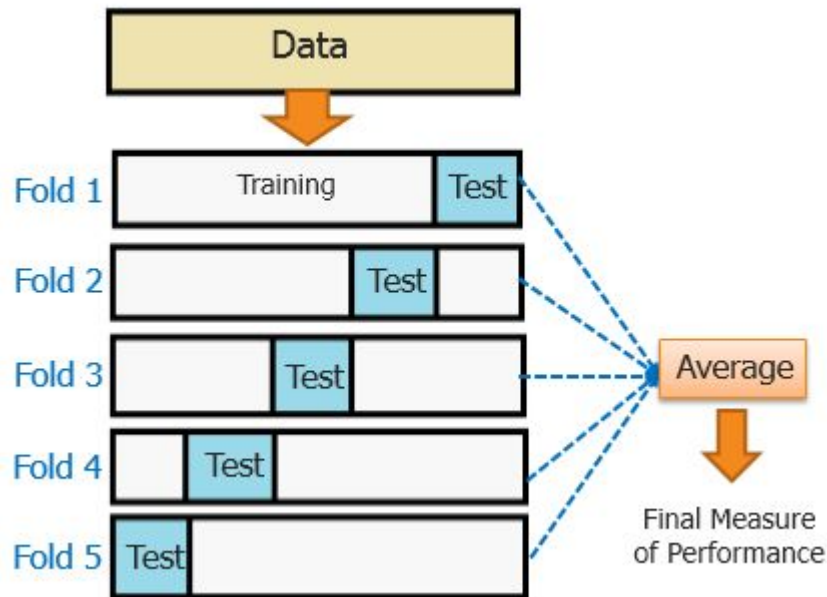
- Разбиваем выборку на k частей
- $k-1$ частей используются для обучения и одна - для тестирования
- Процесс повторяется k раз. Каждый раз для тестирования выбирается разная часть
- Результаты тестирования усредняются

Плюсы:

- Погрешность оценки уменьшается, т.к. используется весь набор

Минусы:

- Обучение производится k раз. Для некоторых моделей это может быть очень долго



Кросс-Валидация

- Плюсы

- Качество измеряется на всем наборе данных
- Качество не зависит от выбора конкретного тестового набора
- Сложнее переобучиться под тест

- Минусы

- Скорость!

- Что выбрать

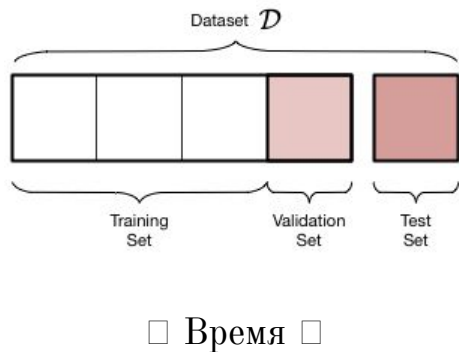
- Мало обучающих данных => Кросс-Валидация
- Много обучающих данных => Train/Validate split

- Не забыть

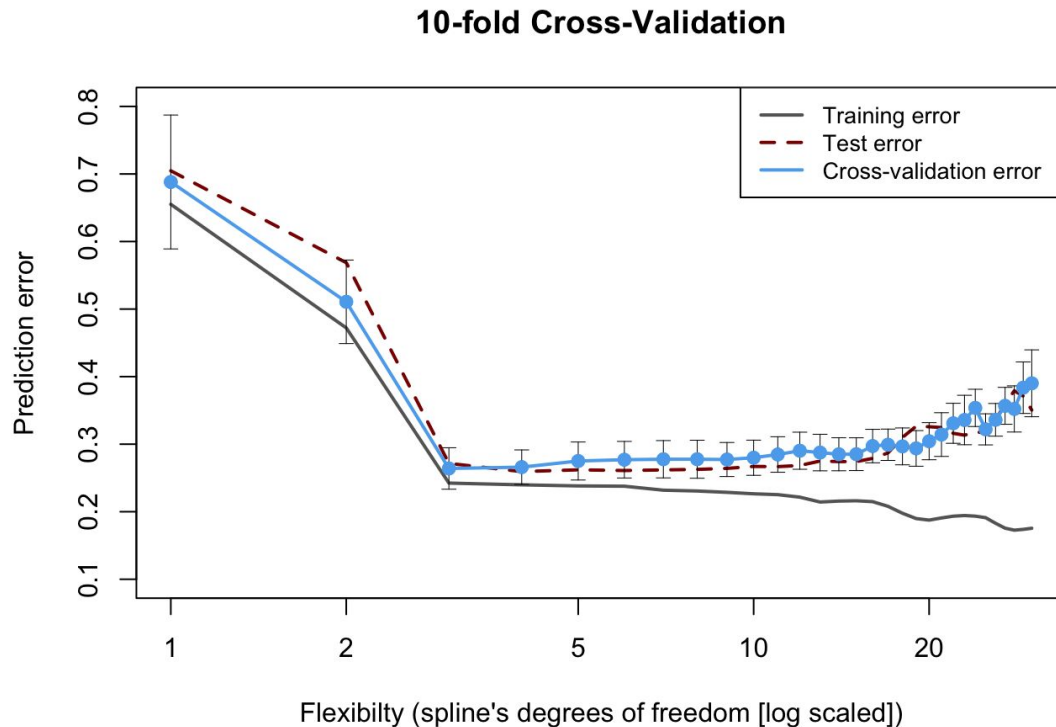
- Отложить Test для замера итогового качества
- Обучить итоговую модель на всех данных

Кросс-Валидация По Времени

- Используется для анализа временных рядов
 - Тестовый набор выбирается из самых свежих данных. Обучение на более старых.
- Полезно в реальных задачах
 - Если в качестве признаков используется множество сигналов, которые могут меняться от времени.
 - Есть возможность определить дату наблюдения.



Кросс-Валидация, пример



Тезисы лекции

- Данные нужно превращать в числа - признаковое описание
- В данных должна присутствовать целевая переменная
- Можно обучить модель предсказывать целевую переменную - это называется обучение с учителем
- Если предсказывается число - это регрессия, если класс - классификация
- Качество модели оценивается с помощью метрик

