ГУАП

Основы машинного обучения

Поляк Марк Дмитриевич

2025

Базовые концепции

Лекция 2

Постановка задачи обучения на примерах

- X – множество *объектов*
- Y – множество *ответов* (предсказаний, оценок, прогнозов)
- $\varphi(x), \varphi: X \to Y$ – неизвестная зависимость (target function)

Дано:

- $\{x_1, \dots, x_\ell\} \subset X$ – обучающая выборка (training sample)
- $y_i = \varphi(x_i), i = 1, \dots, \ell$ – известные ответы

Найти:

- $g(x, \theta), g: X \times \Theta \to Y$ – алгоритм, функция принятия решений или параметрическая модель, приближающая φ на всей выборке X
- $\theta \in \Theta$ – вектор параметров модели, такой, что $g(x, \theta) \approx \varphi(x)$

Обучение на примерах

Весь курс машинного обучения посвящен поиску ответов на следующие вопросы:

- Как задаются (описываются) множества объектов и ответов?
- Насколько точно алгоритм g аппроксимирует целевую функцию φ?
- Как можно оценить алгоритм g?

Описание объектов. Векторы признаков

$f_j: X \to D_j, j = 1, \dots, n$ – признаки объектов (features)

Типы скалярных признаков:

- $D_j = \{0,1\}$ – бинарный признак f_j;
- $|D_j| < \infty$ – номинальный признак f_j;
- $|D_j| < \infty$, D_j упорядочено – порядковый признак f_j;
- $D_j = \mathbb{R}$ – количественный признак f_j: интервал или число.

Вектор $(f_1(x), \dots, f_n(x))$ – *признаковое описание* объекта x.

Матрица признаков: $F = \|f_j(x_i)\|_{\ell \times n} = \begin{pmatrix} f_1(x_1) & \dots & f_n(x_1) \\ \vdots & \ddots & \vdots \\ f_1(x_\ell) & \dots & f_n(x_\ell) \end{pmatrix}$

Описание ответов. Типы задач МО

Задачи обучения с учителем (supervised learning):
- Заданы «ответы учителя» $y_i = \varphi(x_i)$ на обучающих x_i

— задачи классификации (classification):
  - $Y = \{-1, +1\}$ – бинарная классификация (два класса);
  - $Y = \{1, \dots, M\}$ – классификация между M не пересекающимися классами;
  - $Y = \{0,1\}^M$ – M классов, которые могут пересекаться.

— задачи регрессии (regression):
  - $Y = \mathbb{R}$ или $Y = \mathbb{R}^m$.

— задачи ранжирования (ranking):
  - Y – конечное упорядоченное множество.

Задачи обучения без учителя (unsupervised learning)
- Ответов нет, но требуется что-то сделать с самими объектами

Статистическое (машинное) обучение с учителем

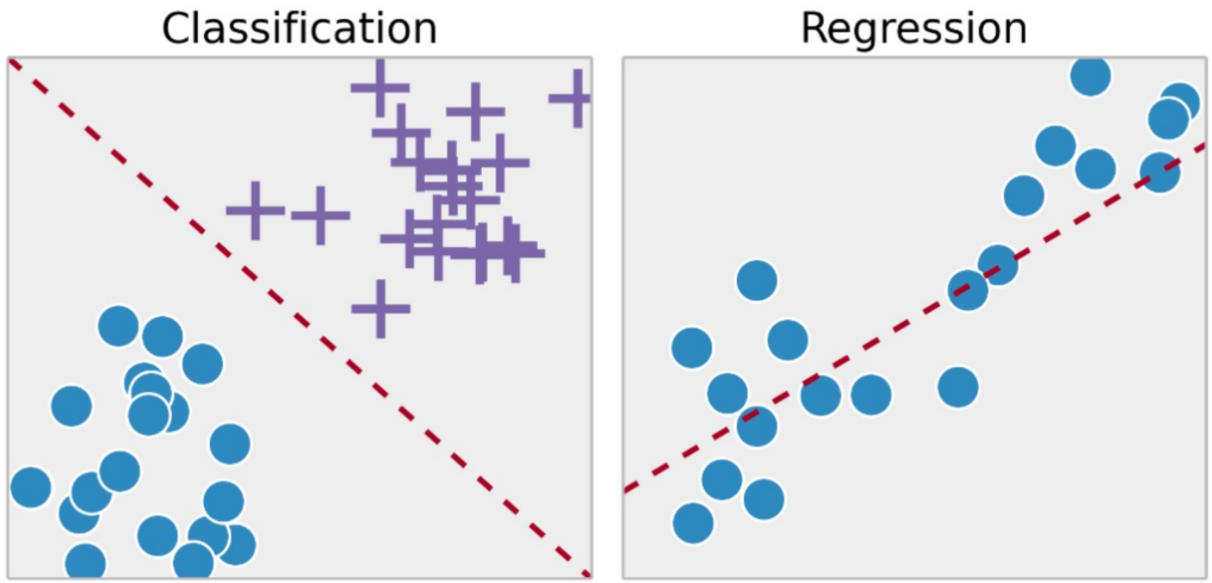
= обучение по прецедентам

= восстановление зависимости по эмпирическим данным

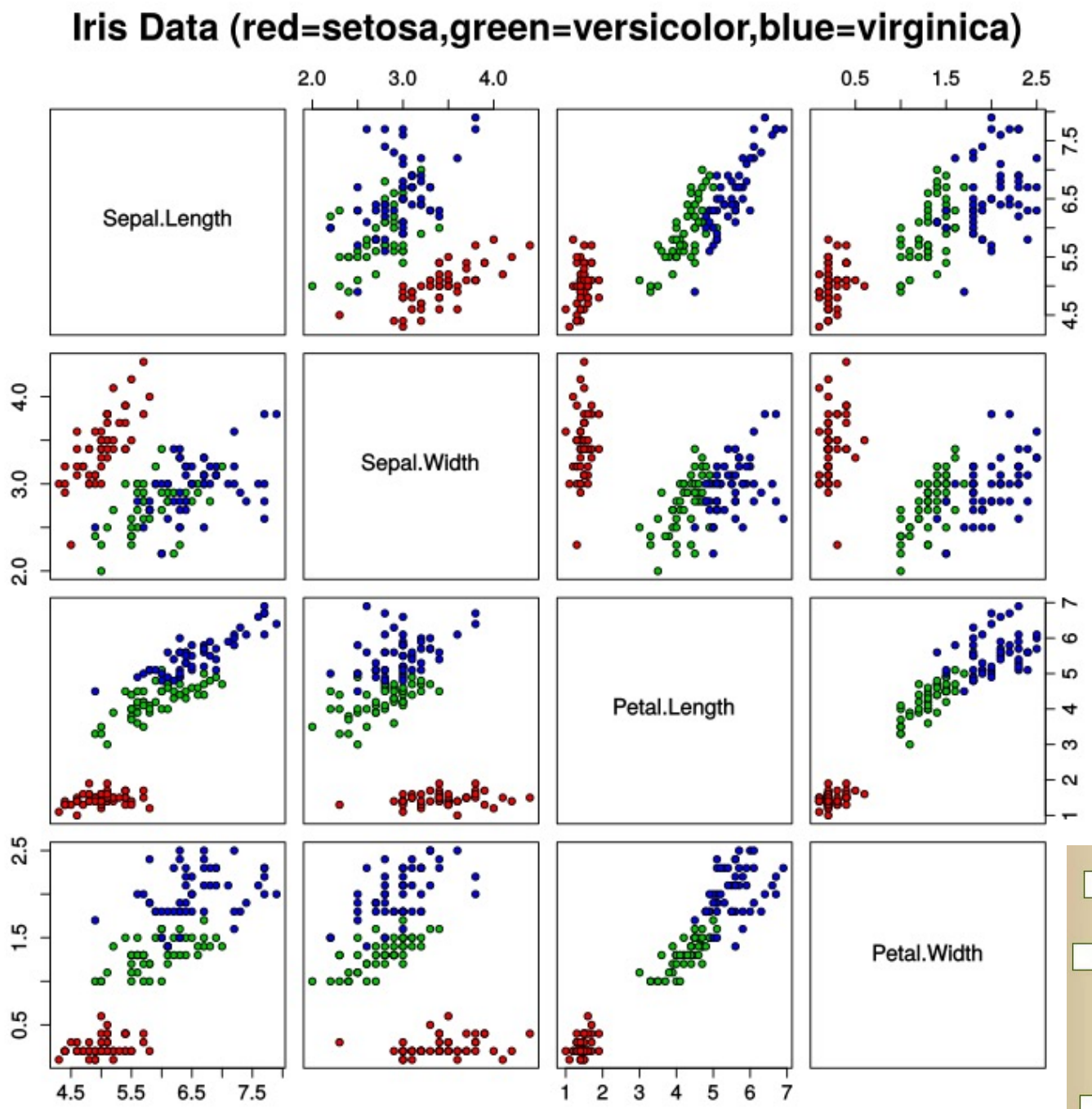
= предсказательное моделирование

= аппроксимация функций по заданным точкам

Два основных типа задач — **классификация** и **регрессия**



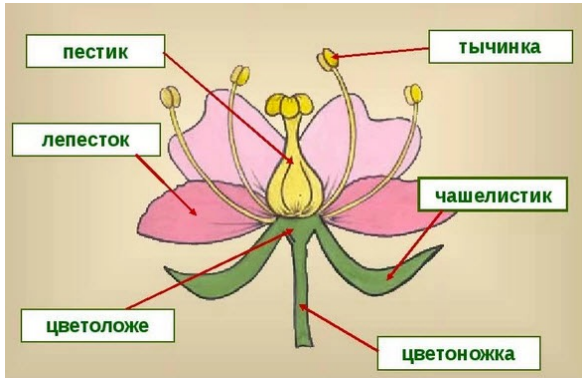
Пример: классификация ирисов Фишера



$n = 4$ признака,
$|Y| = 3$ класса,
длина выборки
$\ell = 150$ объектов

petal – лепесток,
sepal – чашелистик

Данные собраны Рональдом Фишером в 1936 году



Предсказательные модели

Модель (predictive model) — параметрическое семейство функций

$$A = \{g(x, \theta) \mid \theta \in \Theta\},$$

где $g: X \times \Theta \to Y$ – фиксированная функция,
Θ – множество допустимых значений параметра θ.

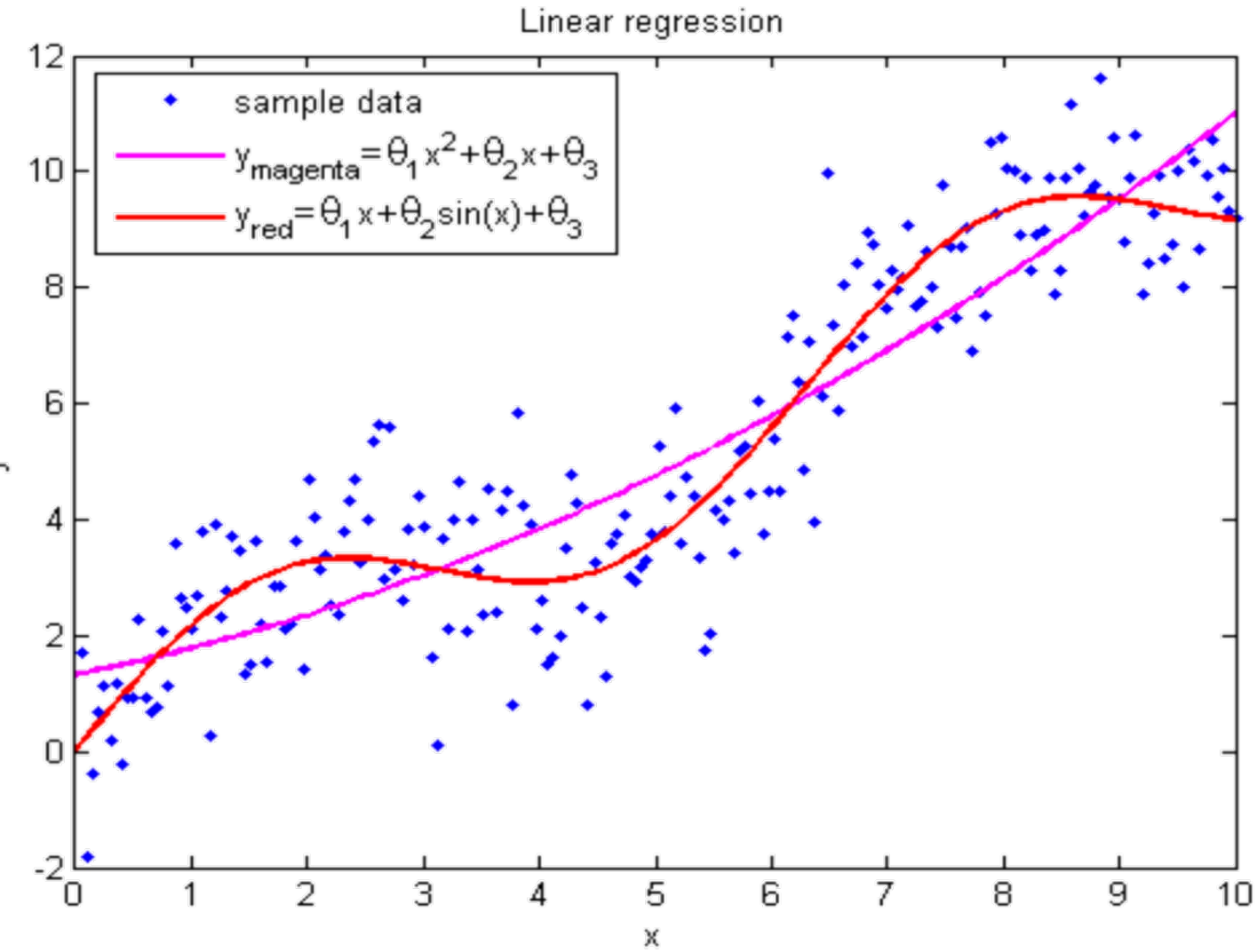
- **Пример**

Линейная модель с векторным параметром $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n) \in \mathbb{R}^n$:

$g(x, \boldsymbol{\theta}) = \sum_{j=1}^{n} \theta_j f_j(x)$ – для регрессии и ранжирования, $Y = \mathbb{R}$;

$g(x, \boldsymbol{\theta}) = \operatorname{sign} \sum_{j=1}^{n} \theta_j f_j(x)$ – для классификации, $Y = \{-1, +1\}$.

Пример: задача регрессии, синтетические данные



$X = Y = \mathbb{R}, \ell = 200,$

$n = 3$ признака:
$\{x^2, x, 1\}$ или $\{x, \sin x, 1\}$

Выводы:

- вычисление новых признаков может улучшить модель
- важно правильно «угадать» модель (подобрать ее форму, т.е. вид функциональной зависимости)

Алгоритм обучения

Процесс обучения с учителем состоит из двух этапов:

- **Обучение** (train):

Алгоритм обучения (learning algorithm) μ: $(X \times Y)^{\ell} \to \Theta$ по выборке $X^{\ell} = (x_i, y_i)_{i=1}^{\ell}$ строит функцию $g(x, \theta)$, оценивая (оптимизируя) параметры модели $\theta \in \Theta$.

- **Применение** (test):

Функция $g(x, \theta)$ для новых объектов x_i' выдает ответы $g(x_i', \theta)$.

Обучение и применение модели

- Обучение

$$\left[\begin{pmatrix} f_1(x_1) & \cdots & f_n(x_1) \\ \vdots & \ddots & \vdots \\ f_1(x_\ell) & \cdots & f_n(x_\ell) \end{pmatrix} \xrightarrow{\varphi} \begin{pmatrix} y_1 \\ \vdots \\ y_\ell \end{pmatrix}\right] \xrightarrow{\mu} \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_n \end{pmatrix} = \boldsymbol{\theta}$$

- Применение

$$\begin{pmatrix} f_1(x'_1) & \cdots & f_n(x'_1) \\ \vdots & \ddots & \vdots \\ f_1(x'_k) & \cdots & f_n(x'_k) \end{pmatrix} \xrightarrow{g} \begin{pmatrix} g(x'_1, \boldsymbol{\theta}) \\ \dots \\ g(x'_k, \boldsymbol{\theta}) \end{pmatrix}$$

Оценивание моделей. Функция потерь

Функция потерь $\mathcal{L}(g, x)$: для заданного объекта $x \in X$ вычисляет величину ошибки алгоритма (функции) $g \in A$ на этом объекте. Ошибка тем больше, чем сильнее $g(x, \theta)$ отклоняется от правильного ответа $\varphi(x)$.

Функция потерь для задач классификации:

- $\mathcal{L}(g, x) = [g(x, \theta) \neq \varphi(x)]$– индикатор ошибки.

Функция потерь для задач регрессии:

- $\mathcal{L}(g, x) = |g(x, \theta) - \varphi(x)|$ – абсолютное значение ошибки;
- $\mathcal{L}(g, x) = \big(g(x, \theta) - \varphi(x)\big)^2$– квадратичная ошибка.

Эмпирический риск

- Нельзя заранее достоверно узнать, на сколько хорошо алгоритм g покажет себя на практике («*риск*»), поскольку неизвестен истинный закон распределения данных $P(x, y)$.
- Оценить и улучшить работу алгоритма g можно на заранее известной ограниченной обучающей выборке (закон больших чисел).
- **Эмпирический риск** – способ оценки качества работы алгоритма g на всей обучающей выборке X^ℓ.
- Функционал эмпирического риска:

$$Q\left(g, X^\ell\right) = \frac{1}{\ell} \sum_{i=1}^{\ell} \mathcal{L}(g, x_i)$$

Замена задачи обучения на задачу оптимизации

Метод минимизации эмпирического риска:

$$\mu\left(X^{\ell}\right) = \arg\min_{g \in A} Q\left(g, X^{\ell}\right)$$

Пример:

- Метод наименьших квадратов ($Y = \mathbb{R}$, $\mathcal{L}$ - квадратичная ошибка)

$$\mu\left(X^{\ell}\right) = \arg\min_{\theta} \sum_{i=1}^{\ell} \left(g(x_i, \theta) - y_i\right)^2$$

Примеры. Задача распознавания месторождений

Объект – геологический район (рудное поле).

Классы – есть или нет полезное ископаемое.

Примеры признаков:

- **бинарные**: присутствие крупных зон смятия и рассланцевания, и т.д.
- **порядковые**: минеральное разнообразие; мнения экспертов о наличии полезного ископаемого, и т.д.
- **количественные**: содержание сурьмы, присутствие в рудах антимонита, и т.д.

Особенности задачи:

- проблема «малых данных» – для редких типов месторождений объектов много меньше, чем признаков.

Примеры. Задача кредитного скоринга

Объект – заявка на выдачу банком кредита.

Классы – bad или good.

Примеры признаков:

- **бинарные**: пол, наличие телефона, и т.д.
- **номинальные**: место проживания, профессия, работодатель, и т.д.
- **порядковые**: образование, должность, и т.д.
- **количественные**: возраст, зарплата, стаж работы, доход семьи, сумма кредита, и т.д.

Особенности задачи:

- нужно оценивать вероятность дефолта $P(y(x) = \text{bad})$.

Примеры. Задача предсказания оттока клиентов оператора мобильной связи

Объект – абонент (клиент) в определенный момент времени.

Классы – уйдет или не уйдет в следующем месяце.

Примеры признаков:

- **бинарные**: корпоративный клиент, включение услуг, и т.д.
- **номинальные**: тарифный план, регион проживания, и т.д.
- **количественные**: длительность разговоров (входящих, исходящих, СМС и т.д.), частота оплаты, и т.д.

Особенности задачи:

- нужно оценивать вероятность ухода;
- сверхбольшие выборки;
- признаки приходится вычислять по «сырым» данным.

Примеры. Задача категоризации текстовых документов

Объект – текстовый документ.

Классы – рубрики иерархического тематического каталога.

Примеры признаков:

- **номинальные**: автор, издание, год, и т.д.
- **количественные**: для каждого термина – частота в тексте, в заголовках, в аннотации, и т.д.

Особенности задачи:

- лишь небольшая часть документов имеют метки y_i;
- документ может относиться к нескольким рубрикам.

Примеры. Задача прогнозирования стоимости недвижимости

Объект – квартира в Санкт-Петербурге.

Предсказать – рыночную стоимость на определенный момент времени.

Примеры признаков:

- **бинарные**: наличие балкона, лифта, мусоропровода, охраны, и т.д.
- **номинальные**: район города, тип дома (кирпичный/панельный/блочный/монолит), и т.д.
- **количественные**: число комнат, жилая площадь, расстояние до центра, расстояние до метро, возраст дома, и т.д.

Особенности задачи:

- выборка неоднородна, стоимость меняется со временем;
- разнотипные признаки;
- для линейной модели нужны преобразования признаков.

Примеры. Задача прогнозирования объемов продаж

Объект – тройка
⟨товар, магазин, день⟩.

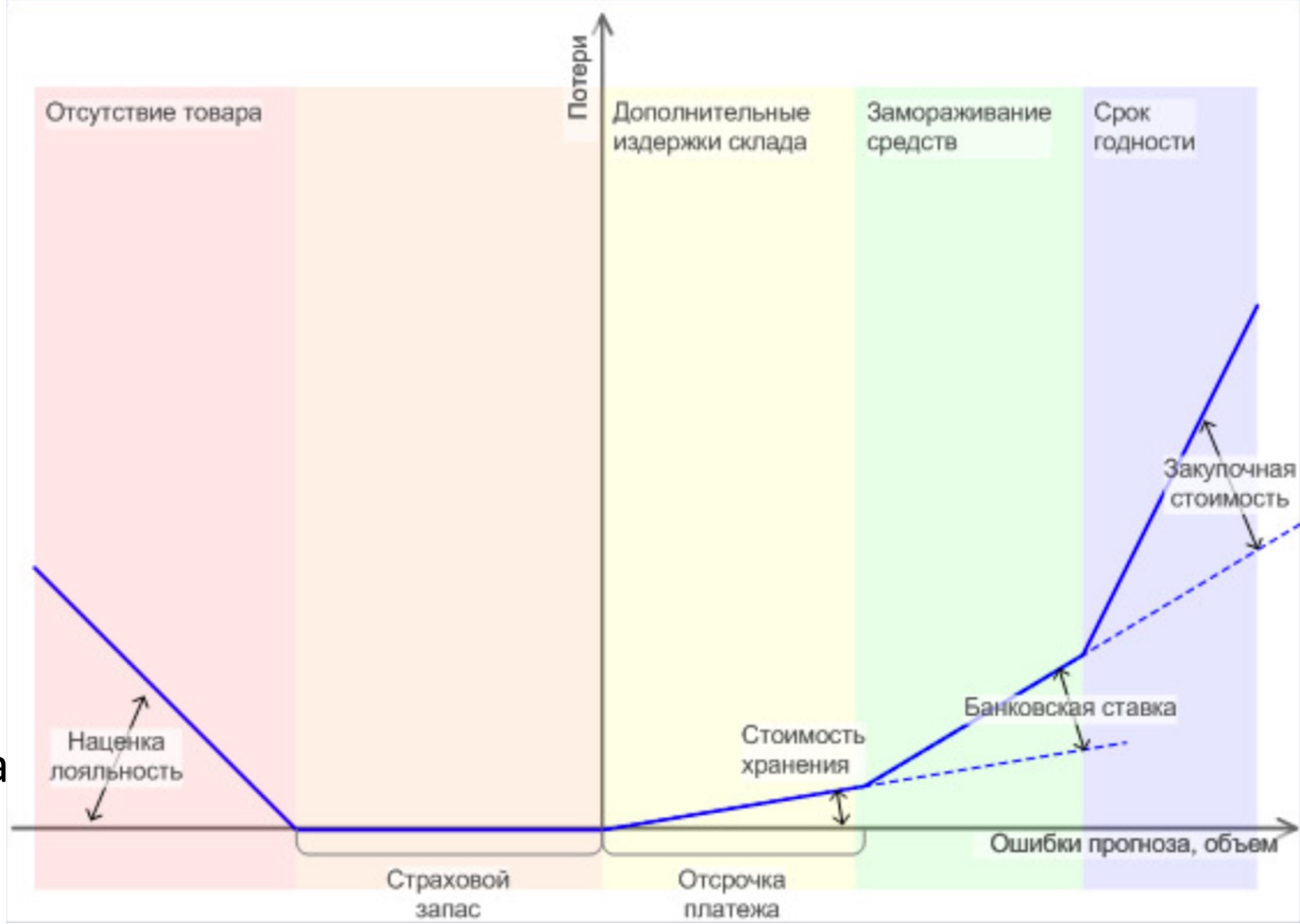
Предсказать – объем продаж в определенном магазине в заданный день.

Примеры признаков:

- **бинарные**: выходной день, праздник, промоакция, и т.д.
- **количественные**: объемы продаж в предыдущие дни.

Особенности задачи:

- функция потерь не квадратична и даже не симметрична;
- разреженные данные.



Примеры. Конкурс kaggle.com: TFI Restaurant Revenue Prediction

Объект – место для открытия нового ресторана.

Предсказать – прибыль от ресторана через год.

Примеры признаков:

- демографические данные: возраст, достаток, и т.д.;
- цены на недвижимость поблизости;
- маркетинговые данные: наличие школ, офисов, и т.д.

Особенности задачи:

- мало объектов, много признаков;
- разнотипные признаки;
- есть выбросы;
- разнородные объекты (возможно имеет смысл строить разные модели для мелких и крупных городов).

https://www.kaggle.com/c/restaurant-revenue-prediction