Экзаменационные вопросы и теормин по курсу

“Методы Машинного Обучения”

группа 521, 2015 год

лекции читали Майсурадзе А. И. (Арчил Иверьевич) и Сенько О. В. (Олег Валентинович)

Экзаменационные вопросы

## 

## **Основы анализа данных (часть Майсурадзе)**

1. Место и роль ИАД в современной структуре человеческой деятельности. Три уровня технологий анализа данных, их назначение. Понятие о моделировании реального мира в науке. Физическая модель. Модель “решателя”. Информационная модель. Эвристическая модель. Основной приём ИАД для связи предметных областей и фундаментальной математики (обучение и эксплуатация эвристической информационной модели).
2. Основные модели данных (dataframe, multidimensional, similarity tensor, transactional). Гомогенные и гетерогенные модели. Фундаментальные задачи ИАД и основные инструменты статистики.
3. Распределение фундаментальных задач ИАД и основных инструментов статистики по моделям данных: в разрезе исходных данных, в разрезе результатов.

## **Основные модели данных (часть Майсурадзе)**

1. Модель данных «признаковое описание объектов». Понятие о шкалах значений атрибутов. Представление реляционными технологиями. Схемы «звезда» и «снежинка».
2. Многомерная модель данных. Группирование объектов как переход к многомерной модели данных. Аналитические пространства. Измерения и категории. Показатели. Детализация. Функции агрегирования, типы показателей по агрегированию.
3. Транзакционная модель данных. Связанные с ней задачи.

## **Классификация как задача (часть Майсурадзе)**

1. Общая задача классификации. Понятие об обучении и использовании. Объект, модель, алгоритм-классификатор. Универсальные ограничения. Локальные ограничения. Оптимизационный подход.
2. Функционалы качества на размеченной выборке. Частотные функционалы качества. Случай бинарной классификации. Стоимостные функционалы качества. Несоответствие частотных и стоимостных функционалов качества человеческому поведению.

~~Подходы к многокритериальной оптимизации. -~~

## **Статистический подход к распознаванию (часть Сенько)**

1. Понятие байесовского классификатора как оптимального алгоритма распознавания.
2. Классификаторы, основанные на использовании формулы Байеса. Линейный дискриминант Фишера.
3. Классификаторы, основанные на использовании формулы Байеса. Логистическая регрессия.
4. Метод k-ближайших соседей.

## **Практические навыки работы с данными (часть Майсурадзе)**

1. Форматы представления информации. Текстовые файлы, их атрибуты, проблема определения атрибутов. Текстовые форматы представления таблиц: separated values, delimited text. Экранирование символов. Форматы представления транзакционных данных.

~~Диаграммы для наборов точек из R^n. -~~

1. Диаграммы для многомерной модели данных. Системы отчётности.

## **Регрессионный анализ (часть Сенько)**

1. Простая регрессия. Множественная регрессия. Поиск коэффициентов по МНК. Недостатки МНК.
2. Трёхкомпонентное разложение ошибки регрессионных моделей.
3. Регуляризация по Тихонову. Гребневая регрессия, лассо, эластичные сети.

## **Метрические тензоры (часть Общая)**

1. Тут везде нужна только постановка задачи кластеризации:

Модель данных «метрические тензоры», гомогенные и гетерогенные многомерные матрицы сходства. Группирование объектов как кластеризация по метрическим описаниям. Гомогенная кластеризация, бикластеризация, мультикластеризация. Основные типы результатов кластеризации (плоская, последовательная плоская, иерархическая, нечёткая, стохастическая, ранговая).

1. Плоская кластеризация. Задача и метод k-means.
2. Последовательная плоская кластеризация. Метод ФОРЕЛЬ.
3. Иерархическая кластеризация. Дивизивная. Агломеративная, функционалы связи (linkage).

~~Задачи точной реализации метрических тензоров. Корректность задачи (разрешимость, однозначность). Алгоритмическая сложность (на примере метрик Минковского). Метрическое многомерное шкалирование, его связь с методом главных компонент.~~

~~Задачи аппроксимации метрических тензоров. Неметрическое многомерное шкалирование. Функционалы стресса. Монотонные (изотонические) отображения, сохранение ранга метрического тензора. Достаточная размерность представления для неразрешимых задач.~~

~~Задача кластеризации как задача аппроксимации метрического тензора. Метрики на метриках, аппроксимация метрик метриками. Метрические деревья. Ультраметрические деревья. Филогенетические деревья, интерпретация длин ребёр и нетерминальных вершин в ультраметрических деревьях, гипотеза молекулярных часов. Гарантированное получение классов эквивалентности. Общая схема вычисления ближайшей ультраметрики.~~

## **Модели и методы ИАД (часть Сенько)**

1. ROC анализ.
2. Линейная модель. Линейная машина как метод обучения линейной модели.
3. Метод опорных векторов.
4. Нейронные сети. Модель перцептрона Розенблатта. Метод его обучения. Теорема Новикова. Переход от сдвига к фиктивному признаку. Многослойные перцептроны. Метод обратного распространения ошибки. Функции активации, удобные для распространения ошибки. Возможность разделения любых множеств.
5. Решающие деревья.
6. Комбинаторно логические методы. Тестовый алгоритм.
7. Основы АВО. Алгоритмы типа КОРА.
8. Алгоритмы, основанные на голосования по наборам закономерностей.

~~Метод логических закономерностей. Метод статистически взвешенных синдромов.~~

1. Bagging. Boosting. Решающие леса (random forest).

~~Коллективные методы. Ошибка и выпуклые комбинации предикторов. Основы алгебраической коррекции.~~

1. Анализ выживаемости как задача.

~~Основы модели Кокса~~.

# Теормин

1. **Понятие о фундаментальных задачах ИАД (основные задачи машинного обучения). Принципы их группирования.**

**Понятие фундаментальный ИАД** - При решении прикладных задач методами ИАД принято делить общую задачу на несколько подзадач (провести декомпозицию), каждая из которых уже известна и изучена. Исторически сложился некоторый набор таких подзадач, на которые удобно проводить декомпозицию. Именно такие задачи называют фундаментальными задачами ИАД.

1. Задачи обучения с учителем:
   1. Задачи классификации - надо получить алгоритм, который может отнести произвольный объект к одному из заранее заданных классов
   2. Задачи восстановления регрессии - надо получить алгоритм (регрессию), который каждому объекту распознавания сопоставит некоторое значение из бесконечного, непрерывного множества
   3. Задачи обучения по прецедентам - классификатор или регрессия настраиваются по заданному конечному набору прецедентов - объектов с заранее известными правильными ответами.
   4. Задачи прогнозирования - обычно прогнозирование сводится к классификации или восстановлению регрессии, когда один из признаков определяет время.
   5. Задачи последовательного обучения - прецеденты приходят последовательно во времени (один за другим). Алгоритм постоянно донастраивается.
   6. Архивирование, настройка модели источника - символы на архивацию приходят последовательно один за другим. (ИАД здесь нужен потому, что нужно уметь предсказывать будущие символы, чтобы кодировать самые популярные наименьшим количеством бит)
2. Задачи обучения без учителя:
   1. Кластеризация (Сегментация) - надо разбить все множество объектов на непересекающиеся подмножества (кластеры, сегменты), в которых объекты в каком-либо смысле похожи друг на друга.
   2. Нечеткая кластеризация, бикластеризация и т.д.
   3. Иерархическая кластеризация (Таксономия) - надо построить дерево подмножеств, в котором каждый последующий слой является измельчением предыдущего.
3. Задачи с частичным обучением - кроме прецедентной информации имеется информация о том, что некоторый набор объектов действительно существует и будет использован в ходе решения прикладной задачи. То есть для настройки алгоритма можно использовать прецедентную информацию и информацию о существовании данных объектов. (например: база данных фотографий, ищем фотографии с лицами)
4. Выявление отклонений, детектирование:
   1. Выявление ошибок в данных - (“так не может быть”). Поступающая информация может содержать ошибки. (например: неисправность измерительного прибора)
   2. Выявление нетипичного поведения - (“так раньше не было”). Наша атомная электростанция раньше никогда не взрывалась. Мы не знаем, как выглядит станция, собирающаяся взорваться, но систему мониторинга создать должны.
   3. Устранение отклонений из обучения (фильтрация) - необходимо выявить те прецедентные данные, которые мешают качественно настроить модель.
5. Задачи восстановления пропусков - решение обычно сводится к задачам классификации или восстановления регрессии
   1. Заполнение пропусков в прецедентах - выбранный метод обучения модели требует, чтобы присутствовали все данные без пропусков.
   2. Заполнение пропусков в описаниях распознаваемых объектов - настроенная модель требует, чтобы во вновь приходящих на обработку описаниях не было пропусков.
6. Анализ наборов (не учитываем время транзакции) Термин: анализ рыночной корзины
   1. Поиск популярных наборов (например: чай и мёд часто покупают вместе)
   2. Поиск ассоциативных правил (например: те, кто купил мёд, часто покупают чай)
7. Анализ последовательностей (учитываем время)
   1. Поиск последовательных правил (например: если сегодня купил принтер, то через месяц купит картридж)
8. Анализ формальных понятий - формализация описания понятия в виде пары (объём, содержание)
   1. Поиск формальных понятий
   2. Построение и анализ решёток понятий
9. Для задач снижения размерности целевой признак отсутствует, в ней переходят от одних описаний к другим. Принято вводить невязку через метрику на пространстве описаний.

Типичный функционал качества агрегирует множество невязок между исходными и новыми описаниями.

1. **Основные модели данных в ИАД. Признаковое описание объекта.**

Данные могут быть гомогенными - т.е. однородными, и гетерогенными, т.е. неоднородными.

Модели для задания исходных данных (в целом все они конвертируемы из одной в другую)

1. **Data Matrix** (признаковое описание) (матрица значений атрибутов каждого объекта) – объекты гомогенные (однотипные).
2. **Multi Dimensional**. - Матрица Кросс-сочетания. – это n-мерный куб, получаемый за счёт декартова произведения матриц с некоторыми атрибутами (например, <simpson vs not simpson x male vs female x child vs adult>)

В каждой ячейке этой матрицы находятся объекты, которые обладают соответствующими признаками.

*Показатель* – это функция от множества объектов из ячейки.

1. **Similarity Tensor** - Используются гетерогенные объекты (т.е. разного типа)

Таблица (аналитическое пространство (тензор)) задаёт взаимодействие между конкретными объектами попарно. (т.е. фактически нам даны оценки похожести разнотипных объектов)

1. **Transactional Data** / **Формальный контекст** (когда носители типа bool) / **Транзакционные данные**

Пусть у предметной области есть понятие «элемент» и «носитель».

И каждому носителю соответствует некоторое количество элементов (называемое «транзакция» - это множество (дубликаты отсутствуют))

1. **Основные модели данных в ИАД. Многомерная модель.**

Основные модели данных в ИАД - см. вопрос 2 теормина

Многомерная модель - **Multi Dimensinal**

Информация в **многомерной модели** *представляется в виде многомерных массивов, называемых гиперкубами*. В одной базе данных, построенной на многомерной модели, может храниться множество таких кубов, на основе которых можно проводить совместный анализ показателей. Конечный пользователь в качестве внешней модели данных получает для анализа определенные срезы или проекции кубов, представляемые в виде обычных двумерных таблиц или графиков.

В клетках этого многомерного куба находятся объекты, обладающие соответствующими признаками

Большинство систем OLAP (Online Analytical Processing) – используют такое представление.

Детализация кросс-таблицы называется Drill Down.

Для показателей (значений ячеек), могут быть функции агрегирования (подсчёт показателя для объединённых ячеек).

Пример полу-агрегируемого показателя – «первый объект множества».

1. **Основные модели данных в ИАД. Многомерная модель. Понятие и видах агрегируемости показателей.**

Основные модели данных в ИАД - см. вопрос 2 теормина

В принципе про агрегируемые показатели написано в предыдущем вопросе, хотя это маловато.

Агрегирование данных (data aggregation): процесс сбора, обработки и представления информации в окончательном виде. Агрегирование данных в основном выполняется для формирования отчетов, выработки политики, управления здравоохранением, научных исследований, статистического анализа и изучения здоровья населения.

Агрегация данных может происходить с использованием нескольких стандартных функций: сумма, минимум, максимум, среднее, количество.

1. **Основные модели данных в ИАД. Тензоры сходства, графы взаимодействия.**

Основные модели данных в ИАД - см. вопрос 2 теормина.

ТЕНЗОРЫ СХОДСТВА что это вообще?

- глянь similarity tensor из 2-го вопроса, что то связанное с этим

- проблема в том, что гугл не знает такого ни (similarity tensor) ни (тензор сходства).

1. **Основные модели данных в ИАД. Транзакционные данные. Формальные контексты.**

Основные модели данных в ИАД - см. вопрос 2 теормина

1. **Фундаментальные задачи ИАД. Задача классификации. Задача восстановления регрессии.**

Фундоментальные задачи ИАД - см вопрос 1 теормина

Тут нужно писать сжато, в вопросе 1 уже написано.

Чтобы во всем этом ЛЕГКО разобраться - первые 4 страницы лекции Майсурадзе:

2.1. Качество классификации (original).pptx

отсюда <https://drive.google.com/folderview?id=0B0X-oQW4pjUUeUY3eE05TDU0Ujg&usp=sharing&tid=0B0X-oQW4pjUUajBvRjVvNVEyMVU#list>

Либо вопрос №7 из билетов

1. **Фундаментальные задачи ИАД. Задача кластерного анализа.**

Фундоментальные задачи ИАД - см вопрос 1 теормина

* Исторически возникла из задачи группировки схожих объектов в единую структуру (кластер) с последующим выявлением общих черт
* В классической задаче кластеризации обучающая выборка представляет собой набор отдельных объектов , характеризующихся вектором вещественнозначных признаков 
* Требуется постросить алгоритм (кластеризатор), который разбил бы выборку на непересекающиеся группы (кластеры) 
* В каждый класс должны попасть объекты в некотором смысле похожие друг на друга

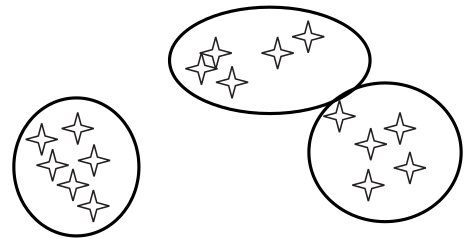


Рис. 1.3. Пример задачи кластеризации. Звездочками обозначены прецеденты. Группы объектов, обведенные кружками, образуют отдельные кластеры

Примеры задач кластерного анализа

• Экономическая география: по физико-географическим и экономическим показателям разбить страны мира на группы схожих по экономическому положению государств

• Финансовая сфера: по сводкам банковских операций выявить группы «подозрительных», нетипичных банков, сгуппировать остальные по степени близости проводимой стратегии

• Маркетинг: по результатам маркетинговых исследований среди множества потребителей выделить характерные группы по степени интереса к продвигаемому продукту

• Социология: по результатам социологических опросов выявить группы общественных проблем, вызывающих схожую реакцию у общества, а также характерные фокус-группы населения

1. **Фундаментальные задачи ИАД. Задача сокращения размерности.**

Фундоментальные задачи ИАД - см вопрос 1 теормина

Для задач снижения размерности целевой признак отсутствует, в ней переходят от одних описаний к другим. Принято вводить невязку через метрику на пространстве описаний.

Типичный функционал качества агрегирует множество невязок между исходными и новыми описаниями.

1. **Обобщающая способность.**

**Обобщающая способность** - точность алгоритма прогнозирования на всевозможных новых, не использованных для обучения объектах, т.е. это точность по всей генеральной совокупности.

**Мерой обобщающей способности** явяется математическое ожидание потерь по всей генеральной совокупности.

Цель задачи прогнозирования - максимизация обобщающей способности. Но нужно понимать, что подсчитать её достаточно проблематично, т.к. обобщающая способность определяется через множество всей генеральной совокупности.

1. **Принцип минимизации эмпирического риска.**

**Эмпирический риск** (Empirical Risk) — это средняя величина ошибки алгоритма на обучающей выборке.

Метод **минимизации эмпирического риска** (Empirical Risk Minimization, ERM) — это общий подход к решению широкого класса задач обучения по прецедентам, в первую очередь — задач обучения с учителем, включая задачи классификации и регрессии.

**Более формально:**

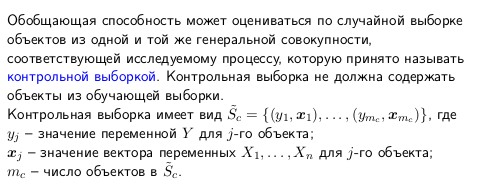
**Эмпирический риск** — это функционал качества, характеризующий среднюю ошибку алгоритма a на выборке X^m:

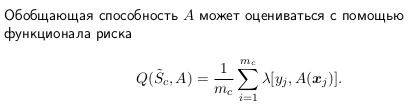
Q(a,X^m) = \frac{1}{m} \sum_{i=1}^m {\mathcal L}(a(x_i),y^{*}(x_i)).

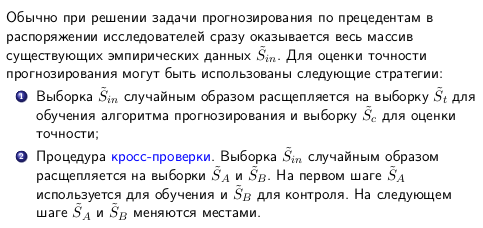
**Метод минимизация эмпирического риска** заключается в том, чтобы в заданной *модели алгоритмов* A найти алгоритм, доставляющий минимальное значение функционалу эмпирического риска:

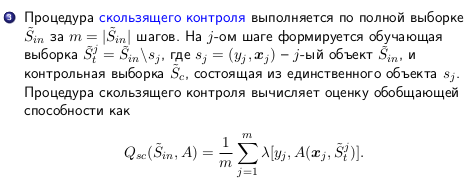
a = \mathrm{arg}\min_{a\in A} Q(a,X^m).

1. **Методы оценивания обобщающей способности. Скользящий контроль.**





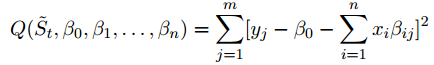


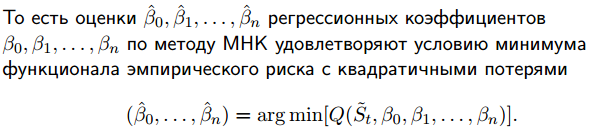


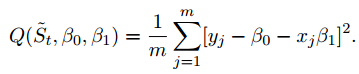
1. **Регрессионный анализ. Метод наименьших квадратов (МНК). Простая одномерная регрессия.**

Распространённым средством решения задач **прогнозирования непрерывной величины *Y* по переменным *X*1 *, . . . , X****n* является использование метода множественной линейной регрессии. В данном методе связь переменной *Y* с переменными *X*1 *, . . . , Xn* задаётся с помощью линейной модели ***Y* = *β*0 + *β*1 *X*1 + *. . .* + *βnXn* + *ε****,* где ***β*0*, β*1*, . . . , βn***- вещественные регрессионные коэффициенты, *ε* - случайная величина, являющаяся ошибкой прогнозирования.

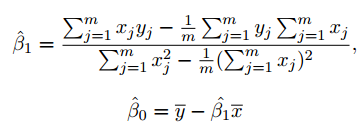
Традиционным способом поиска регрессионных коэффициентов является метод наименьших квадратов (МНК). МНК заключается в **минимизации функционала эмпирического риска с квадратичными потерями**:





Рассмотрим простейший вариант линейной регрессии, описывающей связь между переменной *Y* и единственной переменной *X* : *Y* = *β*0 + *β*1 *X* + *ε* Функционал эмпирического риска на выборке ˜*St* = *{*( *y* 1 *, x* 1) *, . . . ,* ( *y m, x m*) *}* принимает вид:

Необходимым условием минимума функционала *Q*(*S*˜*t, β*0*, β*1) является равенство 0  и . Отсюда:



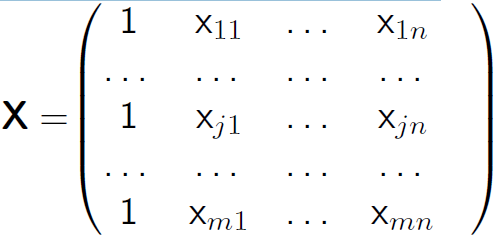
Выражение для *β*ˆ1 может быть переписано в виде:

, где *Cov*(*Y, X | S*˜*t*) является выборочной ковариацией переменных *Y* и

*X*, *D*(*X | S*˜*t*) является выборочной дисперсией переменной *X.*

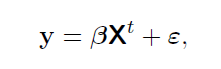
1. **Множественная линейная регрессия. Недостатки МНК.**

При вычислении оценки **вектора** параметров B = (B0, …, Bn) в случае многомерной линейной регрессии удобно использовать матрицу плана X размера m × (n + 1) , которая строится по обучающей выборке ˜ St. **Матрица** плана имеет вид

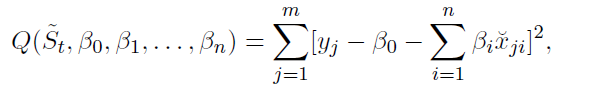


То есть j-я строка матрицы плана представляет собой вектор значений переменных X1, … ,Xn для объекта sj c одной добавленной слева компонентой, содержащей 1.

Пусть y = (y1, … ,ym) - вектор значений переменной Y . Связь Y с переменными X1, … , Xn на объектах обучающей выборки может быть описана с помощью матричного **уравнения**



где e= (e1, … ,em) - вектор ошибок прогнозирования для объектов ˜ St. Функционал Q( ˜ St; B0, B1 , … , Bn) может быть записан в виде



Одним из главных недостатков МНК является устойчивость этого метода, что плохо сказывается в некоторых случаях на обобщающей способности. Причина низкой устойчивости - мультиколлинеарность. Данное явление возникает при сильной коррелированности одной из переменных выборки с какой-либо линейной комбинацией других переменных. (Слайды 16-17 3 Лекция Сенько)

1. **Использование методов регуляризации.**

Регуляризация используется для избавления МНК от недостатков (в плане устойчивости).

Для этого включают дополнительную шумовую компоненту в исходный оптимизируемый функционал.

Типы регуляризаций:

1. **регуляризация по Тихонову** - добавление штрафной компоненты к оптимизируемому функционалу (по формулам там была модификация значений признаков объектов)
2. **гребневая регрессия** (ridge) - добавляет сумму квадратов регрессионных компонентов к функционалу (это приводит к положительности дискриминанта и улучшается устойчивость)
3. **метод лассо** - добавляет модули регрессионных компонент к функционалу (делает отбор переменных (т.к. некоторые регрессионные переменные приравниваются к 0) и применение на маленьких выборках возможным) (при высокой корреляции некоторых переменных, на практике метод лассо ухудшает свои показатели)
4. **эластичная сеть** - добавление суммы квадрата регрессионных компонент с коэффициентом θ и модуля регрессионных компонент с коэффициентом (1-θ) (вбирает все лучшее из первых двух).
5. **Методы распознавания. Что такое байесовский классификатор? Байесовские методы в распознавании.**

**Распознавание** — это отнесение исходных данных к определенному классу с помощью выделения существенных признаков, характеризующих эти данные, из общей массы несущественных данных.

**Байесовский классификатор**.

Относит объект в тот класс, для которого условная вероятность P(класс i | x) максимальна. (при решении как правило нужно считать по формуле Байеса).

Байесовский подход к классификации основан на теореме, утверждающей, что если плотности распределения каждого из классов известны, то искомый алгоритм можно выписать в явном аналитическом виде. Более того, этот алгоритм оптимален, то есть обладает минимальной вероятностью ошибок.

1. **Метод k-ближайших соседей.**

Алгоритм классификации объектов. Оценка условных вероятностей P(K i | x) ведётся по ближайшей окрестности точки x, содержащей k признаковых описаний объектов обучающей выборки (k соседних объектов). В качестве оценки выступает отношение ki/k , где ki - число признаковых описаний объектов обучающей выборки из класса Ki внутри рассматриваемой окрестности точки x (*т.е. в итоге объект присваивается тому классу, который является наиболее распространённым среди соседей данного элемента*). Окрестность задаётся с помощью функции расстояния ρ(x, x) заданной на декартовом произведении X×X , где X - область допустимых значений признаковых описаний. В качестве функции расстояния может быть использована стандартная эвклидова метрика или метрика Хэмминга в случае с бинарными признаками. Единственным параметром для настройки является число соседей (k).

1. **Комбинаторно-логические методы . Тупиковый тест. Представительный набор.**

(Тут надо полагаться на покрытия и тесты, представлением элементов вектором признаков)

**Принцип частичной прецедентности**. При распознании диагностике важны некоторые комбинации (а именно конкретные наборы, а не целые классы)

**Тест** – это набор признаков, позволяющих разделить классы. (т.е. любые 2 объекта из любых 2-х классов отличаются хотя бы по одному признаку из теста)

**Тупиковый тест** – тест, из которого нельзя удалить ни один из признаков.

Выбор класса при классификации происходит на основании голосования по тесту (учёт совпавших признаков, причём могут учитываться какие-то веса)

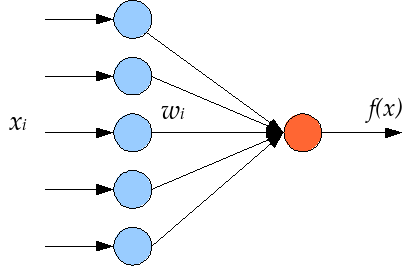
Поиск теста – это задача о покрытии, NP-трудная.

**Алгоритм типа Кора** – когда ищется не только набор признаков, но набор признаков со своим описанием.

**Представительный набор** – набор описания (зафиксированные признаки), которого присутствуют только в одном классе.

Из существования теста следует существование набора, но не наоборот.

1. **Алгоритм вычисления оценок.**
2. Выделяется множество подмножеств, которое называется «**система опорных векторов**».
3. Функция близости сравнивает 2 объекта по некоторому опорному множеству. (например =1 если все признаки совпали, или =0 если нет (а может учитываться какой-то порог, …))
4. **Модель нейрона. Перцептрон Розенблатта.**



**Нейрон** - некоторая сущность, которая суммирует входящие сигналы с учётом весов, применяет к результату активационную функцию и выдаёт результат на выход.

Сигналы, приходящие на вход перцептронов рецепторов интерпретируются как входные признаки.

3 типа нейронов:

1. нейроны-рецепторы (сенсоры)
2. внутренние нейроны - имеет множество входных связей от рецепторов или внутренних нейронов
3. реагирующие нейроны - имеет множество входных связей от рецепторов или внутренних нейронов

Все сигналы, входящие в нейрон обрабатываются: z = w0 + ∑wiui , после чего применяется активационная функция Ф(z) - и это значение становится результатом нейрона и выдаётся другим нейронам (один сигнал с выхода может передаваться многим другим на вход) (ui - значения с других нейронов, wi - веса внутри нейрона, w0 - параметр сдвига)

**Перцептрон Роззенблатта** (1957) - нейросетевая модель с одним единственным нейроном

Используется для задачи распознавания с 2-мя классами, активационная функция Ф=1, если z>=0, Ф= -1, если z<0

Особенность перцептрона - простая и эффективная процедура обучения (вычисления wi)

Настройка параметра происходит по обучающей выборке. (предварительно признаковое описание преобразуется в векторные сигналы, а затем вектора описаний из класса K2 умножаются на -1, а из K1 не меняются)

1. **Многослойный перцептрон. Методы обучения.**

**Многослойный перцептрон** - нейросетевые методы распознавания, задаваемые комбинациями связанных между собой нейронов.

Обладает существенно более высокой аппроксимирующей способностью

Обычно сеть формируется в виде слоёв: слои внутренних нейронов осуществляют преобразование сигналов. Слой реагирующих нейронов производит окончательную классификацию объектов на основании сигналов, поступающих от нейронов, принадлежащих внутренним слоям.

Обычно соблюдаются следующие правила формирования структуры:

1. Допускаются связи между только между нейронами, находящимися в соседних слоях.
2. Связи между нейронами внутри одного слоя отсутствуют.
3. Активационные функции для всех внутренних нейронов идентичны

Для решения задачи распознавания с L классами используется конфигруация с L реагирующими нейронами.

Сигналы вычисляемые на выходе реагирующих нейронов, интерпретируются как оценки за классы.

Процедура обучения перцептрона (одного):

1. Случайно выбирается нулевое приближение вектора весовых коэффициентов
2. Преобразованные описания обучающей выборки последовательно передаются на вход перцептрона
3. Если писание x(k), поданное на k-м шаге классифицируется неправильно, то происходит коррекция w(k+1) = w(k) + x(k), если классификация верна, то ничего не меняется.
4. Повторяем до тех пор, пока:
   1. достигается полное разделение объектов из классов K1 и K2
   2. повторение подряд заданного числа операций не приводит к улучшению разделения
   3. оказывается исчерпанный заранее заданный лимит итераций

Процедура обучения многослойного перцептрона - **метод обратного распространения ошибки**:

**Потери классификации** объекта будем считать как сумму квадрата разности между выходом реагирующего нейрона и тем, что он должен был выдать (т.е. 0, когда объект не принадлежит классу, и 1, иначе)

**Качество аппроксимации** на обучающей выборке - это сумма потерь для каждого объекта выборки. (веса фиксированны). Цель - подобрать такие веса, чтобы улучшить качество аппроксимации.

Основа обучения - метод градиентного спуска.

Метод градиентного спуска - оптимизирует произвольный функционал F(θ), θ(k) = θ(k-1) + n \* grad(F(θ(k-1)))

n - парамер, задающий размер каждого шага (**learning rate**)

Процедура обучения - такая же как и для одного перцептрона, за исключением того, как именно изменяются веса. Они меняются в соответствии с методом градиентного спуска: там простыми расчётами выводится градиент, и потом оно от нижнего слоя к верхнему по цепочке может распространяться, при этом для подсчёта градиента - сложность линейна.

1. **Метод опорных векторов. Основные понятия. Возможность нелинейного разделения.**

**Основная идея метода** — перевод исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с максимальным зазором в этом пространстве.

Две параллельных гиперплоскости строятся по обеим сторонам гиперплоскости, разделяющей наши классы. **Разделяющей гиперплоскостью** будет гиперплоскость, максимизирующая расстояние до двух параллельных гиперплоскостей. Алгоритм работает в предположении, что чем больше разница или расстояние между этими параллельными гиперплоскостями, тем меньше будет средняя ошибка классификатора.

1. **Методы решающих деревьев и решающих лесов.**

**Decision tree** cредство поддержки принятия решений, использующееся в статистике и анализе данных для прогнозных моделей.

На ребрах («ветках») дерева решения записаны атрибуты, от которых зависит целевая функция, в «листьях» записаны значения целевой функции, а в остальных узлах — атрибуты, по которым различаются случаи. Чтобы классифицировать новый случай, надо спуститься по дереву до листа и выдать соответствующее значение.

При анализе решений «**дерево решений**» используются как визуальный и аналитический инструмент поддержки принятия решений, где рассчитываются ожидаемые значения (или ожидаемая полезность) конкурирующих альтернатив.

**Random forest** (решающий лес) - алгоритм машинного обучения, заключающийся в использовании комитета (ансамбля) решающих деревьев. Алгоритм сочетает в себе две основные идеи: метод бэггинга Бреймана, и метод случайных подпространств.

1. **Методы построения коллективных решений в распознавании.**

Используются **ансамбли** распознающих алгоритмов и далее на основе определенных комитетов принимается окончательное коллективное решение.

В данном контексте для построения ансамблей используются методы **бэггинг** (метод, относящий объект в тот класс, куда его отнесло большинство алгоритмов -- высокий прирост обобщающей способности (можно учитывать веса)) и **бустинг** (последовательное наращивание ансамбля, в котором объекты выбираются не равноправно, а исходя из какого-то вероятностного распределения).

Важно что ошибка взвешанного решения не больше, чем сумма взвешанных ошибок. Т.е. коллективное решение всегда даёт результат - не хуже.

На паре говорили, что бустниг (bootstrap aggregating) - это изменение обучающей выборки, путём замены случайных объектов.

А бэггинг - это идея повышения веса объектов, которые были неправильно распознаны.

1. **Основы алгебраической коррекции.**

Задача распознавания в алгебраической теории рассматривается как задача построения по начальной информации I о классах K1, . . . , KL для предъявленной для распознавания выборки

Sc = {s1 , . . . , sq } правильной информационной матрицы ∥αj ∥L×q .

/\* or for dummies \*/

Вспомним, что классификация это ROC, т.е. R \* C, где R – это оператор, ставящий каждому прецеденту некоторую характеристику – вероятность отнесения к каждому из классов. А С – это оператор, решающий по характеристикам к какому классу, будем принадлежать.

Журавлёв предложил ввести операции:

1. Почленное умножение матриц
2. Почленное сложение матриц
3. Почленное умножение на число

Тогда S (матрица, где столбец – прецендент из обучающей выборки) \* R = Г для каждого классификатора нашего ансамбля можно рассматривать как набор. Для которого существует замыкание из множества всех матриц Г, которые получаются из 1-3 пункта. Тогда существует в этом наборе Гcorr - **корректная матрица** (матрица, для которой корректно работает стандартное решающее правило).

В итоге надо в замыкании найти корректную матрицу.

1. **Задача анализа выживаемости (краткое определение)**

Анализ выживаемости - позволяющих оценить вероятность наступления события. Это математическая модель зависимости функции риска от независимых переменных-факторов. В анализе выживаемости решается задача оценки функции выживания или функций, производных от нее.

Выбор метода для анализа выживаемости зависит от исходных данных:

* наличия зависимой и независимых переменных-предикторов;
* наличия или отсутствия цензурированных данных.

Из слайдов: **Целью таких задач** является восстановление вероятности того, что ожидаемое критическое событие с исследуемым объектом произойдёт не ранее произвольного момента времени. Таким критическим событием может быть отказ изделия в технике, гибель испытуемого организма в биологии или смерть пациента в медицине. Таким образом целью анализа является вычисление функции (кривой) выживаемости S(t) = P{T > t} , где через T обозначено время наступления критического события, P {T > t} обозначает вероятность того, что критическое событие произойдёт позже момента t.