Курс “Основы машинного обучения”
Вопросы к контрольной работе

1. Что такое объекты и признаки в машинном обучении? Для чего нужен функционал ошибки? Что такое алгоритм (модель)?
   Объекты x - это то, для чего делается предсказание (абстрактные сущности). Признаки (факторы, features) x=(x1,x2,…, xd)- это характеристики объектов. Функционал ошибки Q(a, X) нужен, чтобы определить меру качества работы алгоритма на всей выборке (например, с помощью MSE).
   Алгоритм (модель) a(x) - это функция, предсказывающая ответ для любого объекта (дающая оценку целевой переменной).
2. Чем задача классификации отличается от задачи регрессии? Приведите примеры задач классификации и регрессии.
   Обе задачи получают обучающую выборку, однако регрессия предсказывает числовые ответы, а классификация выбирает их из конечного множества. Например, прогноз стоимости квартир или определение возможного итогового балла студента (регрессия), определение заболевания или определение кампуса студента (классификация).
3. Чем отличаются задачи бинарной классификации, многоклассовой классификации, классификации с пересекающимися классами?
   Они отличаются областью ответов, бинарная классификация отвечает на вопрос да или нет {0, 1}, многоклассовая выбирает из множества {1, 2, …, n}, а классификация с пересекающимися классами возвращает вектор из 0 и 1 {0, 1}^n, потому что в ней объекты могут принадлежать нескольким классам одновременно.
4. Что такое вещественные (числовые), бинарные, категориальные признаки? Приведите примеры.
   Вещественные признаки Dj=R характеризуют объект числом, их можно сравнивать, складывать и т.п. (например, площадь квартиры или ее расстояние до метро).
   Бинарные признаки Dj={0, 1} определяют, соответствует ли объект признаку или нет (например, квартира новая? (0 - нет, были жильцы; 1 - да, только построили) или пол (0 - мужской; 1 - женский)).
   Категориальные признаки Dj - неупорядоченное множество представляет собой набор значений объекта, которые можно сравнить только на равенство, нельзя сложить или определить больше или меньше какие-то из них (например, города, в которых расположены кампусы НИУ ВШЭ).
5. В чём заключается обобщающая способность алгоритма машинного обучения? К чему приводит её отсутствие? Что такое переобучение?
   Обобщающая способность заключается в способности алгоритма выдавать правильные результаты не только для объектов, участвовавших в обучении, но и для новых, которые не входили в обучающую выборку. Её отсутствие приводит к переобучению алгоритма, когда модель слишком адаптировалась к объектам из обучающей выборке и выдает на ней очень хорошее качество, но на тестовой оно низкое.

6. Что такое гиперпараметр? Чем гиперпараметры отличаются от обычных параметров алгоритмов? Приведите примеры параметров и гиперпараметров в линейных моделях.
   Гиперпараметр - это параметр алгоритма, который нельзя подбирать по обучающей выборке. Гиперпараметры отличаются от обычных параметров алгоритма тем, что параметры можно подбирать по обучающей выборке, поскольку они отвечают за подстройку модели под данные, а гиперпаметры, наоборот, мешают модели подстраиваться под обучающую выборку. Например, число ближайших соседей или коэффициент регуляризации.
7. Что такое отложенная выборка? Что такое кросс-валидация (скользящий контроль)? Как ими пользоваться для выбора гиперпараметров?
   Отложенная выборка - это деление выборки на обучающую и тестовую обычно в пропорции 70/30 или 80/20.
   Кросс-валидация (скользящий контроль) - это разбиение данных на k блоков (обычно 3 или 5, еще есть классный, но медленный leave-one-out с k=1), где каждый из блоков по очереди становится тестовым набором данных.
   Для выбора гиперпараметров можно запустить обучение на них с разными значениями k (числа соседей) и выбрать то значение, при котором качество модели выше.
8. Как метод k ближайших соседей определяет класс для нового объекта?
   После обучения модели объекты обучающей выборки сортируются в порядке возрастания расстояния от нового объекта, после чего выделяются k ближайших объектов и на вычисляется, сколько объектов из k ближайших совпадают с новым, после чего берется самый популярный класс.
   $$a(x) = argmax \sum_{i=1}^{k} [y = y(i)]$$
9. Опишите метод k ближайших соседей с парзеновским окном. Какие в нём есть параметры?
   Метод k ближайших соседей с парзеновским окном применяется для решения задачи классификации, с помощью него можно при выборе k ближайших ориентироваться на веса, определять, какое расстояние далекое, а какое близкое. Его параметрами являются K() и h, где K() - это ядро (обычно используется гауссовское), а h - ширина окна (обычно используют 0,5).
10. Запишите формулу метода kNN для регрессии.
    $$a(x) = \frac{1}{k} \sum_{i=1}^{k} y(i)$$
11. Что такое градиент? Какое его свойство используется в машинном обучении?
    Градиент - это вектор частных производных. В машинном обучении он используется для того, чтобы определять, в какую сторону функция быстрее всего растет (или уменьшается, если градиент отрицательный). При помощи градиента MSE можно определить оптимальный вектор-весов w (но это долго и периодически проблемно).
12. Опишите алгоритм градиентного спуска.
    Стартуем от любой точки x0 и двигаемся вниз по антиградиенту до тех пор, пока не достигнем минимума.
    Более подробно: сперва инициализируем веса (например, генерируем случайные числа для этого), затем считаем до сходимости для каждой точки разность между текущим значением и произведением размера шага и градиента в

предыдущей точке. Процесс останавливается при соблюдении определенного условия (1 - разница между точками очень маленькая; 2 - найденный градиент очень маленький; 3 - ошибка на отложенной выборке перестала уменьшаться).

13. Что такое стохастический градиентный спуск? В чём его отличия от обычного градиентного спуска? Какие у него плюсы и минусы?
    Стохастический градиентный спуск - это метод поиска минимума функции, который использует случайные объекты вместо суммы всех объектов.
    Его отличие от обычного градиентного спуска заключается в том, что он использует только один случайно выбранный обучающий объект за итерацию, а не все из них.
    Его главное преимущество - быстрота спуска, а недостаток - низкое качество на промежуточных шагах.
14. В чём отличия обучения линейной регрессии с помощью аналитической формулы и с помощью градиентного спуска?
    Решение с аналитической формулой долгое и может не иметь конечного результата, если матрица вырожденная или близка к ней. Градиентный спуск может разойтись и не выдать итоговый результат из-за этого. Так, если обратная матрица существует и градиентный спуск сходится (дошёл до точки с нулевым градиентом), результат с аналитической выдаст оптимальный вектор весов, а градиентный спуск выдаст приближенный вектор с какой-то погрешностью, но полученные наборы весов будут очень близки друг к другу.
15. Что такое регуляризация? Как она помогает бороться с переобучением?
    Регуляризация - это система штрафования за большие веса коэффициентов и сложность модели, используется для предотвращения слишком сильной подгонки модели к обучающей выборке. Обеспечивает более гладкую и устойчивую модель, которая лучше обобщает закономерности и уменьшает риск переобучения, сохраняя адекватный уровень смещения.
16. Чем L1-регуляризация отличается от L2-регуляризации?
    L-1 регуляризация (Lasso) использует для системы штрафования больших весов абсолютные значения коэффициентов, при этом незначимые признаки обнуляет.
    L-2 регуляризация (Ridge) использует сумму квадратов коэффициентов, работает с мультиколлериальностью.
17. Что такое масштабирование признаков? Как его проводить? Зачем это нужно?
    ***Масштабирование признаков** - приведение признаков к одинаковой размерности с помощью смещения и сжатия данных относительно нуля, чтобы можно было сравнивать веса между собой для лучшей интерпретации модели. Нужно из каждого признака вычесть среднее значение и поделить на стандартное отклонение.*
18. Чем функционал MSE отличается от MAE? Что такое функция потерь Хубера и для чего она нужна?
    MSE использует квадраты отклонений, MAE - абсолютные значения разниц между реальными и предсказанными значениями, поэтому MSE более чувствительна к выбросам, в отличие от MAE. Для градиентного спуска удобнее использовать MSE, MAE требует большей оптимизации, кроме того, иногда задачи требует штрафование за большой разброс. Функция потерь Хубера - это совмещение MSE и MAE в функции потерь так, чтобы для малых ошибок она вела себя как MSE, а для больших ошибок как MAE. Она нужна для

минимизации влияния выбросов на линию регрессии с сохранением общей точности модели.

19. Как выглядит модель линейной классификации в случае двух классов?
    Модель берет знак от скалярного произведения <w, x>, т.е. a(x) = sign(<w,x>)
20. Что такое отступ? Для чего он нужен?
    Отступ - это показатель, насколько модель права Mi=yi<w,xi>.
    Нужен для того, чтобы определить правильно ли классификатор определяет класс (если Mi>0, правильно, иначе нет) и чтобы оценить уверенность в определении класса (чем больше модуль отступа, тем лучше, это происходит за счет удаленности объекта от разделяющей гиперплоскости).
21. Как обучаются линейные классификаторы (общая схема с верхними оценками)?
    !!!!!!Еще раз посмотреть, не до конца понимаю - 8 лекция!!!!!!!!!!
22. Для чего может понадобиться оценивать вероятности классов?

23. Как обучается логистическая регрессия? Запишите функционал и объясните, откуда он берётся.

24. Как в логистической регрессии строится прогноз для нового объекта?

25. Что такое калибровочная кривая?

26. Что такое метод опорных векторов? Опишите его основную идею. На что влияет гиперпараметр C?

27. Чем отличаются функции потерь в логистической регрессии и в SVM?

28. Как устроены метрики accuracy, precision, recall? Что такое F-мера? Чем она лучше арифметического среднего точности и полноты?
    Accuracy =
    Precision =
    Recall =
    F-мера - Она лучше арифметического среднего точности и полноты тем, что F-мера будет низкой, если хотя бы одна из двух метрик низкая.
29. Для чего нужны ROC- и PR-кривые? Как они строятся? Что такое AUC-ROC и AUC-PRC?

30. Как можно свести задачу многоклассовой классификации к серии задач бинарной классификации?

31. Что такое микро- и макро-усреднение?

32. Что такое решающее дерево? Как оно строит прогноз для объекта? Что такое предикат, какие предикаты вы знаете?
    Решающее дерево - это модель, разделяющая решающую поверхность на области в зависимости от условия, состоит из внутренних вершин с предикатами и листьев с прогнозами из области ответов.

Если строится прогноз для классификации объекта, то берется наиболее часто встречаемый в области класс. Если строится прогноз для регрессии, то значение рассчитывается как средний ответ.
Предикаты - это условия для разбиения выборки, могут быть следующими: порог на признак [xj<t], с линейной моделью [<w,x> < t], с метрикой [p(x, x0)<t] и т.п.

33. Как обучаются решающие деревья в задачах классификации и регрессии? Что такое impurity (критерии хаотичности)? Какие вы знаете критерии для регрессии и для классификации?
    Обучаются жадно, при построении для каждого уровня выбирается лучший предикат (дает максимальное изменение хаотичности), далее рекурсивно повторяем для дочерних вершин, до выполнения критерия остановки.
    impurity (характеристика помешанности классов) =
    Для классификации используется критерий информативности:
    Для регрессии энтропия рассчитывается с помощью дисперсии:
34. Какие вы знаете критерии останова и способы выбора значений в листьях? Какие гиперпараметры имеются у деревьев?
    Критерии останова могут быть следующими: 1 - ограничение глубины; 2 - ограничение количества листьев; 3 - установка минимального числа объектов по итогу; 4 - установка минимального уменьшения хаотичности при разбиении.
    Способы выбора значений в листьях: самый частый класс или среднее значение.
    Гиперпараметры деревьев: максимальная глубина, кол-во элементов в листе.
35. Что такое бэггинг и метод случайных подмножеств?
    Бэггинг - это композиция моделей, обученных независимо на случайных подмножествах объектов из выборки (т.е. случайная подвыборка объектов).
    Метод случайных подмножеств - это композиция моделей, обученных по случайному набору признаков (т.е. случайное подмножество признаков).
36. Что такое случайный лес, как он обучается и как он строит прогнозы? В чём его отличия от обычного бэггинга над деревьями?
    Случайный лес - это метод построения разнообразных и независимых друг от друга базовых моделей решающих деревьев на основе бэггинга. Перед обучением выбирается число деревьев n, с помощью бутстрапа (случайная выборка с возвращением) генерируется выборка, по которой строится решающее дерево до тех пор, пока количество его объектов в листе не более n_min, оптимальное разбиение ищется среди q случайно выбранных (с помощью метода случайных подмножеств) признаков, причем при каждом разбиении они выбираются заново.
    Для построения прогнозов ?? !!!!!!Еще раз посмотреть, не до конца понимаю - лекция!!!!!!!!!!
    Отличия от обычного бэггинга над деревьями: случайный лес применяет случайные подпространства для выбора признаков, в то время как бэггинг использует все доступные признаки.
37. В чём идея разложения ошибки на смещение и разброс? Как бэггинг меняет смещение и разброс одной модели?
    Ошибка состоит из 3 слагаемых: шум (сложность и противоречивость собранных данных), смещение (близость модели к идеалу, т.е. достаточно ли сложная модель), разброс (устойчивость модели к изменениям в обучающей выборке).
    Бэггинг ??!!!!!!Еще раз посмотреть, не до конца понимаю - лекция!!!!!!!!!!

38. Опишите идею градиентного бустинга для среднеквадратичной ошибки. Запишите задачу для обучения очередной базовой модели.
    Для среднеквадратичной ошибки бустинг сводится к обучению с заменой целевой переменной на остатки.
    $\frac{1}{l}\sum_{i=1}^{l}(bn(xi) - (yi - a(n-1)(xi)))^2 -> min\, bn(x)$
39. Опишите идею градиентного бустинга для произвольной функции потерь. Запишите задачу для обучения очередной базовой модели.

    $\frac{1}{l}\sum_{i=1}^{l}(bn(xi) - (yi - a(n-1)(xi)))^2 -> min\, bn(x)$
40. Какие модели обычно используют в качестве базовых в градиентном бустинге? Почему?

41. Для чего в бустинге используют сокращение шага? Как оно устроено?

42. Что такое leaf-wise обучение дерева? А что такое oblivious decision tree?

В машинном обучении смещение (bias) и разброс (variance) — это две ключевые составляющие ошибки модели. Они меняются в зависимости от типа алгоритма:

1. Дерево решений (Decision Tree)
- **Смещение**: Низкое (если дерево глубокое).
  - Дерево может идеально подогнаться под обучающие данные (переобучение).
- **Разброс**: Высокий (если дерево глубокое).
  - Малейшие изменения в данных могут сильно изменить структуру дерева.
- **Как меняется?**
  - При увеличении глубины дерева:
    - Смещение ↘ (уменьшается, т.к. модель точнее описывает данные).
    - Разброс ↗ (увеличивается, т.к. модель становится чувствительнее к шуму).
  - При уменьшении глубины:
    - Смещение ↗ (модель слишком простая, недообучается).
    - Разброс ↘ (модель менее чувствительна к данным).

---

2. Бэггинг (Random Forest — "Лес" деревьев)
- **Смещение**: Примерно как у одного дерева (но обычно лучше из-за усреднения).
- **Разброс**: Сильно уменьшается по сравнению с одним деревом.
- **Как меняется?**
  - Чем больше деревьев в лесе:
    - Смещение почти не меняется (остаётся низким).
    - Разброс ↘ (уменьшается за счёт усреднения предсказаний).
  - Чем больше признаков в случайном подмножестве (`max_features`):
    - Разброс ↗ (деревья становятся более похожими).
    - Смещение может немного ↘ (если признаки важные).

---

3. Бустинг (Gradient Boosting, XGBoost, LightGBM, CatBoost)
- **Смещение**: Постепенно уменьшается.
- **Разброс**: Может увеличиваться, но контролируется регуляризацией.
- **Как меняется?**
  - Чем больше итераций (количество деревьев):
    - Смещение ↘ (каждое новое дерево исправляет ошибки предыдущих).
    - Разброс ↗ (если бустинг переобучается).
  - Чем больше глубина деревьев (`max_depth`):
    - Смещение ↘ (модель становится сложнее).
    - Разброс ↗ (риск переобучения).
  - Чем сильнее регуляризация (`learning_rate`, `lambda`, `gamma`):
    - Смещение ↗ (модель становится более консервативной).
    - Разброс ↘ (предсказания становятся устойчивее).

Выводы
Алгоритм	Смещение (Bias)	Разброс (Variance)
Дерево	Низкое (если глубокое)	Высокий
Случайный лес	Среднее/Низкое	Низкий
Бустинг	Очень низкое	Средний/Высокий (зависит от регуляризации)

- **Лес уменьшает разброс**, усредняя множество деревьев.
- **Бустинг уменьшает смещение**, последовательно улучшая предсказания.
- **Одно дерево** склонно к переобучению (высокий разброс).

Оптимальный выбор зависит от данных: если шумов много — лучше лес, если нужно точное предсказание — бустинг с аккуратной настройкой.