Маширов Даниил ПИ22-2

**САМОСТОЯТЕЛЬНАЯ РАБОТА**

**ПО СОСТАВЛЕНИЮ ТЕСТОВЫХ ВОПРОСОВ**

**ML0.1**

1) Метод библиотеки NumPy, который используется для вычисления среднего значения элементов массива называется:

1. mean() ~%0%
2. average() ~%0%
3. median() ~%0%
4. mean(axis=None) ~%100%
5. avg() ~%0%

2) Метод библиотеки Pandas, который используется для объединения двух DataFrame по общему столбцу называется:

1. merge('column\_name') ~%0%
2. merge(on='column\_name') ~%100%
3. join('column\_name') ~%0%
4. combine('column\_name') ~%0%
5. concat('column\_name') ~%0%

**ML0.2**

1) Метод библиотеки Pandas, который используется для группировки данных по одному или нескольким столбцам и последующего применения агрегирующей функции к каждой группе называется:

1. group\_by() ~%0%
2. groupby() ~%100%
3. aggregate() ~%0%
4. summarize() ~%0%
5. combine\_by() ~%0%

2) Метод библиотеки Pandas, который используется для удаления дубликатов из DataFrame называется:

1. drop\_duplicates() ~%100%
2. remove\_duplicates() ~%0%
3. delete\_duplicates() ~%0%
4. clean\_duplicates() ~%0%
5. drop\_repeated() ~%0%

**ML1.1**

1) Правильным утверждением относительно задачи регрессии является:

1. Регрессия используется для решения задач классификации. ~%0%
2. Задача регрессии связана с прогнозированием числовых значений. ~%100%
3. Регрессия решает задачу кластеризации данных. ~%0%
4. Метод градиентного спуска применяется только в задачах регрессии. ~%0%
5. Регрессия не требует обучения на размеченных данных. ~%0%

2) Методы, которые широко используются для минимизации функции потерь в задачах регрессии – это:

1. Метод исключения переменных. ~%0%
2. Метод квадратичного программирования. ~%0%
3. Метод градиентного спуска. ~%50%
4. Метод случайного леса. ~%0%
5. Метод опорных векторов. ~%0%
6. Метод наименьших квадратов. ~%50%

**ML1.2**

1) Правильным утверждение относительно множественных регрессионных моделей с использованием библиотеки sklearn является:

1. Множественная регрессия применяется только для категориальных данных. ~%0%
2. Множественные регрессионные модели могут использоваться только для прогнозирования дискретных значений. ~%0%
3. Множественная регрессия позволяет моделировать зависимость множества независимых переменных от зависимой переменной. ~%100%
4. sklearn не поддерживает множественные регрессионные модели. ~%0%
5. Множественная регрессия используется только в области компьютерного зрения. ~%0%

2) Методы, которые можно использовать для оценивания качества множественной регрессионной модели в библиотеке sklearn – это:

1. Использование метода evaluate() у экземпляра модели. ~%0%
2. Применение метода accuracy\_score() к предсказаниям модели. ~%0%
3. Использование метода score() у экземпляра модели с передачей тестовых данных. ~%100%
4. Применение метода classification\_report() к результатам модели. ~%0%
5. Использование метрик, таких как средняя абсолютная ошибка (MAE) или среднеквадратичная ошибка (MSE). ~%100%
6. Использование метода predict() и сравнение результатов с истинными значениями. ~%0%

**ML1.3**

1) Верным утверждением относительно моделей регрессии в библиотеке scikit-learn является:

1. Модели регрессии используются только для категориальных данных. ~%0%
2. Модели регрессии предназначены исключительно для классификации. ~%0%
3. Модели регрессии в scikit-learn могут прогнозировать непрерывные числовые значения. ~%100%
4. scikit-learn не поддерживает модели регрессии. ~%0%
5. Модели регрессии применяются только в области обработки изображений. ~%0%

2) Метрики, которые могут быть использованы для оценки качества моделей регрессии в scikit-learn – это:

1. accuracy\_score() ~%0%
2. classification\_report() ~%0%
3. mean\_absolute\_error() ~%100%
4. precision\_score() ~%0%
5. r2\_score() ~%100%
6. confusion\_matrix() ~%0%

**ML2.1**

1) Основным шагом в процессе обучения модели логистической регрессии в scikit-learn является:

1. Загрузка данных, применение модели без обучения. ~%0%
2. Разделение данных на обучающий и тестовый наборы. ~%100%
3. Применение модели без указания признаков. ~%0%
4. Обучение модели без использования оптимизаторов. ~%0%
5. Оценка качества модели до ее обучения. ~%0%\

2) Функцией, которая используется для принятия решений в логистической регрессии является:

1. Гиперболический тангенс. ~%0%
2. Сигмоидная функция. ~%100%
3. Ректификатор. ~%0%
4. Линейная функция. ~%0%
5. Экспоненциальная функция. ~%0%

**ML2.2**

1) Вариантом, который описывает, как влияет несбалансированность классов на процесс обучения моделей классификации является:

1. Несбалансированность классов не оказывает влияния на процесс обучения. ~%0%
2. Модели классификации всегда лучше работают с несбалансированными классами. ~%0%
3. Несбалансированные классы могут привести к смещению модели в пользу чаще встречающегося класса, что делает ее менее способной к выявлению редких классов. ~%100%
4. Несбалансированные классы ускоряют процесс обучения. ~%0%
5. Модели классификации не могут работать с несбалансированными данными. ~%0%

2) Верными утверждениями о границах принятия решений в моделях классификации являются:

1. Граница принятия решений представляет собой область, разделяющую классы. ~%33%
2. Форма границы принятия решения всегда зависит от вида используемой модели. ~%0%
3. Данные всегда являются линейно разделимыми. ~%0%
4. Логистическая регрессия, будучи линейным методом, хорошо справляется с линейно разделимыми данными. ~%33%
5. Введение полиномиальных признаков в модель может помочь, если данные линейно неразделимы. ~%33%
6. Граница принятия решений никогда не может быть прямой линией. ~%0%