Федеральное государственное бюджетное образовательное

учреждение высшего образования

**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»**

**Департамент анализа данных и машинного обучения**

Пояснительная записка к курсовой работе

по дисциплине

«Технологии анализа данных и машинное обучение»

на тему:

**«**Сравнение методов регрессии на реальных наборах данных**»**

Выполнил:

студент группы ПИ19-4

факультета информационных

технологий и анализа больших данных

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Кузнецов М. А.



Научный руководитель:

канд.физ.-мат.н., доцент

Романова Е. В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Москва**

**2022**

**Содержание**

[Введение 3](#_Toc102132369)

[1. Постановка задачи 4](#_Toc102132370)

[2. Описание предметной области 5](#_Toc102132371)

[3. Анализ данных 6](#_Toc102132372)

[3.1. Описательный анализ 6](#_Toc102132373)

[4. Обучение моделей с помощью машинного обучения. 14](#_Toc102132374)

[4.1. Линейная регрессия 14](#_Toc102132375)

[4.2. Множественная регрессия 16](#_Toc102132376)

[4.3. Полиномиальная регрессия 18](#_Toc102132377)

[4.4. Случайный лес 21](#_Toc102132378)

[4.5. Деревья решений 26](#_Toc102132379)

[4.6. Регрессия LASSO 28](#_Toc102132380)

[4.7. Гребневая регрессия 29](#_Toc102132381)

[4.8. Метод k ближайших соседей 31](#_Toc102132382)

[5. Выбор модели 36](#_Toc102132383)

[5.1. Grid Search 36](#_Toc102132384)

[6. Анализ полученных результатов 42](#_Toc102132385)

[Заключение 44](#_Toc102132386)

[Ссылка на репозиторий с полным кодом выполнения работы 45](#_Toc102132387)

[Список использованных источников 46](#_Toc102132388)

# Введение

С каждым годом рост популярности информационных технологий все больше и больше затрагивает различные сферы деятельности. Компаниям, ученым, исследователям, а также, студентам приходится работать с достаточно большим количеством информации. Такой объем необходимо оценить, и найти закономерности, на основе которых можно будет создать некоторые сервисы или программы. Но человек может уставать, и из-за этого он, вероятно, допустит, большое количество ошибок. Чтобы избежать такиe ситуации, было придумано использовать мощности компьютера для решения определенных задач, эту науку назвали машинным обучением.

Что же такое машинное обучение? Это наука о том, чтобы компьютер решал определенные задачи, как будто, думая, как человек. Схожесть обучения компьютера и человека действительно существует и понятна – человек ходит в школу, получает знания, становится эрудированным, умным в разных сферах деятельности, а машина обучается на наборе данных, и благодаря обучению, вырабатывает некоторые закономерности, предсказывая разные методы решения задач, постоянно улучшая свое обучение.

В наше время почти каждый человек сталкивался с машинным обучением, но далеко не все знают и понимают, что это, для чего это необходимо, и на что способно. С помощью данной науки было создано множество различных сервисов и программ, таких как, навигаторы, системы распознавания лиц, различные беспилотные автомобили, а также, различные диагностики в сфере медицины. Сейчас, машинное обучение затронуло и используется почти в каждой сфере, от банков, до сельского хозяйства.

Задачи машинного обучения делятся на обучение с учителем и обучение без учителя. Учителем является человек, который имеет «правильные», или же, реальные данные, на основе которых, можно предсказать различные значения, например цену квартиры, по различным факторам, или же, параметрам. Задача регрессии является хорошим примером обучения с учителем. Машинное обучение без учителя предполагает то, что машина будет учиться самостоятельно, не имея «правильных» данных, обычно, задачами такого обучения является сгруппировать разные данные на определенное количество категорий. Задача кластеризация является отличным примером обучения без учителя.

Задачи, которые решают с помощью машинного обучения, можно отнести к одной из следующих категорий [3]:

1. Задача регрессии – вещественный ответ предсказывается на основании различных признаков.
2. Задача кластеризации – данные разбиваются на различные группы категорий, называемые кластерами.
3. Задача классификации – категориальный ответ предсказывается на основании различных признаков.
4. Задача выявления аномалий – отделение аномалий от не аномалий, стандартных случаев.
5. Задача уменьшения размерности – большое количество признаков уменьшаются до более меньшего для последующей визуализации.

# Постановка задачи

Целью данной курсовой работы является сравнение методов регрессии на реальном наборе данных. Алгоритмы машинного обучения описывают обучение некой функции, которая будет соотносить входные переменные с выходной переменной. Задача регрессии относится к классу задач обучения с учителем, так как необходимо спрогнозировать целевую переменную по заданному количеству различных признаков. Методы регрессии является одними из самых распространенных в машинном обучении, также, они считаются эффективными, из-за своего легкого использования и понимания.

Существует достаточно большое количество видов регрессии, каждая из которых имеет свои достоинства и недостатки [2]. В рамках данной курсовой работы будет исследовано восемь различных моделей. Так же, в работе будет предоставлена оценка и сравнение моделей, в результате чего будет выбрана самая точная и эффективная. Все результаты моделирования будут отражены в наглядном виде, а именно, с помощью графиков, и таблиц сравнения моделей.

В качестве данных был выбран датасет, в котором находятся данные об океанографических исследованиях, собираемых более шестидесяти лет. Задачей исследования будет являться оценка влияния солености воды в граммах соли на килограмм воды (г/кг) на температуру воды в градусах Цельсия. В данной работе будет произведено предсказание температуры воды по солёности на основе различных моделей регрессий, таких как:

1. Линейная регрессия
2. Полиномиальная регрессия
3. Множественная регрессия
4. Случайный лес
5. Деревья решений
6. Регрессия LASSO
7. Гребневая (Ридж-регрессия)
8. Метод k ближайших соседей.

# Описание предметной области

В данной работе предметной областью является задача исследования влияния солёности на температуру воды, в ходе чего, можно будет определить, и ответить на вопросы о том, существует ли взаимосвязь ними, а также, получится ли предугадать температуру воды на основе солености.

Данное исследование было выбрано не случайно. Вода является главным фактором существования жизни, Мировой океан занимает более семидесяти процентов поверхности Земли, в нем живут и находятся огромное количество живых организмов, так же, он помогает поглощать более тридцати процентов углекислого газа, а также вырабатывает более пятидесяти процентов кислорода, необходимого человеку. Нельзя не согласиться с тем фактом, что океаны, моря – это легкие планеты, без которых не было бы существования. В дополнении, необходимо подчеркнуть, что Мировой океан хранит в себе огромное количество разных ресурсов, таких как растительных, энергетических, минеральных, и животных, с их помощью истощающиеся запасы суши могут пополниться.

Но, к сожалению, с каждым годом человек все больше и больше истощает запасы морей и океанов, многие ученые бьют тревогу в связи с повышением температуры океанов и морей, которые являются следствием глобального потепления, причиной которой является деятельность человека. Глобальное потепление негативно влияет на океанические воды, влечет за собой непоправимые последствия, такие как закисление океана, таяние ледников, изменение ареалов рыб, и других морских животных, вследствие чего, им приходится переселяться и искать другие места обитания. Так же, повышение температуры океана негативно влияет на человека, из-за чего люди начинают страдать от различных заболеваний, которые были вызваны этой катастрофой.

# Анализ данных

Перед началом работы, связанной с обучением моделей, необходимо провести очистку, а также, предварительный анализ данных. Все действия, сделанные мной, будут показаны в виде скриншотов кода, с подробным описанием.

## Описательный анализ

Проведем подробный анализ признаков датасета, начнем с их описания. Так как целью данной работы является исследование влияния солености на температуру воды, то выделим самые необходимые нам признаки, такие как:

1. T\_degC – Температура воды в градусах Цельсия.
2. Salnty – Солености воды в граммах соли на килограмм воды.
3. Depthm – Глубина воды.

Разделим их на результативные и факториальные признаки: Salnty, Depthm будут результативными, а T\_degC факториальным. Так же, под моделью линейной регрессии будем понимать модель вида (См. Рисунок 1):



Рисунок 1 – модель линейной регрессии

где y – предсказанное значение, x1, …, xk – объясняющие ряды, b1, bk – коэффициенты модели, e – вектор ошибок модели. [1]

В начале работы необходимо загрузить необходимые модули из различных библиотек (См. Рисунок 2).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2 - Загрузка необходимых модулей

После загрузки нам необходимо отобразить датасет, отображать его будем с помощью широко используемой структуры данных – Pandas DataFrame [5]. С помощью метода read\_csv загрузим текстовые данные, разделенными запятыми в DataFrame, а также, введем параметр index\_col, который переименует столбец индексов на Btl\_Cnt, потому что можно представить каждую строку датасета в виде нумерованных бутылок, в каждой из которых находятся результаты отборов проб (См. Рисунок 3).



Рисунок 3 - Чтение данных

Выберем только нужные столбцы из датасета, а также переименуем их названия на более понятные (См. Рисунок 4).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4 – Выбор и переименование столбцов

Произведем подготовительную предобработку данных. Так как наборы данных часто могут содержать пропущенные значения, а обучение модели с большим количеством пропусков может значительно повлиять на качество модели машинного обучения, необходимо подобрать алгоритм нахождения и замены таких значений. Существует несколько способов замены:

1. Вменение с использованием самых часто встречающихся значений. Подходит, как и для числовых, так и для категориальных типов данных.
2. Замена пустых значений на средние значения или медианы по столбцам. Подходит только для числовых типов данных.
3. Заполнение пропущенного значения последним не пропущенным. Подходит только для числовых типов данных.
4. Замена пустых значений на моду по столбцам. Подходит для бинарных признаков.

В данной работе все числовые данные были заменены на способ №3 (Заполнением последним не пропущенным значением), а все категориальные типы данных заменены на слово None (См. Рисунок 5).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 5 - Замена пустых значений.

Функция, выполняющая получение и чтение данных, переименование столбцов, а также, обработку пропущенных значений, возвращающая обработанный датасет (См. Рисунок 6). Вызываем эту функцию, показываем первые 5 строк для наглядности (См. Рисунок 7).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 6 - Функция чтения и предобработки данных

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 7 – Отображение первых пяти строк датасета.

Детальная сводка числовых значений (См. Рисунок 8). Данный метод возвращает такие показатели, как среднее значение, стандартное отклонение, минимум и максимум, квантили, а также, количество переменных в наборе данных.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 8 - Детальная сводка

Выводим попарную корреляцию всех столбцов в кадре данных (См. Рисунок 9).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 9 - Попарная корреляция

Визуализируем данные с помощью парных графиков (См. Рисунок 10).

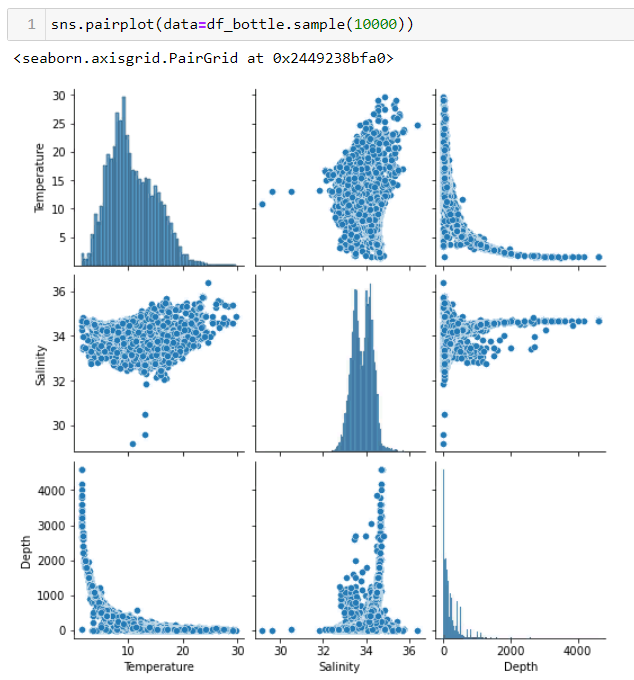


Рисунок 10 - Графическое представление данных

Построим корреляционную матрицу. Корреляция – это мера взаимосвязи переменных друг с другом для того, чтобы оценить, как одна переменная изменяется по отношению к другой (См. Рисунок 11) [4].

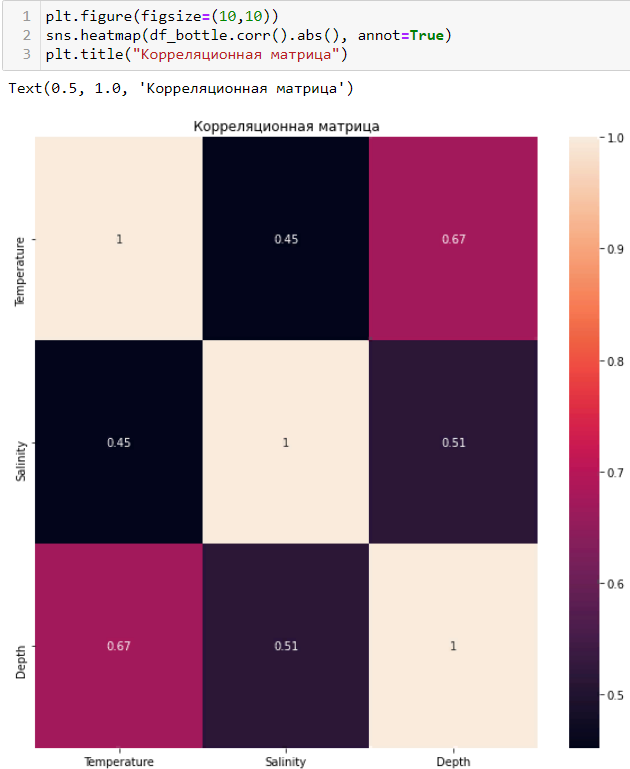


Рисунок 11 - Корреляционная матрица

Для оценки качества и эффективности модели используются метрики, так же, по показателям результатов метрик можно сравнить разные модели, определить, какая из них делает более точный прогноз. Мы будем использовать несколько метрик, расскажем про каждую из них.

Среднеквадратическая ошибка – показатель для оценки регрессии, который служит для того, чтобы измерить среднеквадратическую ошибку прогнозов. Чем выше это значение, тем хуже модель. Данный показатель определяется формулой (См. Рисунок 12), где yi – ожидаемый результат, а ŷi – прогноз модели.

Изображение выглядит как текст, часы

Автоматически созданное описание

Рисунок 12 - Формула средней квадратической ошибки

Средняя абсолютная ошибка – показатель для оценки регрессии, измеряет среднее расстояние между предсказанным и истинным значением у модели [6]. Чем выше значение данной ошибки, тем хуже модель. Приведем формулу для расчета средней абсолютной ошибки (См. Рисунок 13).

Изображение выглядит как текст, часы

Автоматически созданное описание

Рисунок 13 - Формула средней абсолютной ошибки

Коэффициент детерминации R^2 является важным показателем для оценки регрессии, показывает точность верного прогноза, или же, функциональную зависимость между переменными. Данный показатель принимает значения от 0 до 1, чем выше к 1, тем лучше модель. Считается, что если коэффициент детерминации больше 80%, то модель достаточно хорошая. Приведем формулу для расчета данного показателя (См. Рисунок 14).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 14 - Формула коэффициента детерминации

# Обучение моделей с помощью машинного обучения.

Перед началом обучения разделим наш набор данных на обучающую и тестовую выборки в соотношении 80/20. Данные столбца температуры передадим в качестве целевого вектора y, а данные солености воды передадим в вектор признаков x (См. Рисунок 15). В тестовую выборку попадет 172973 строки, а в обучающую 691890.

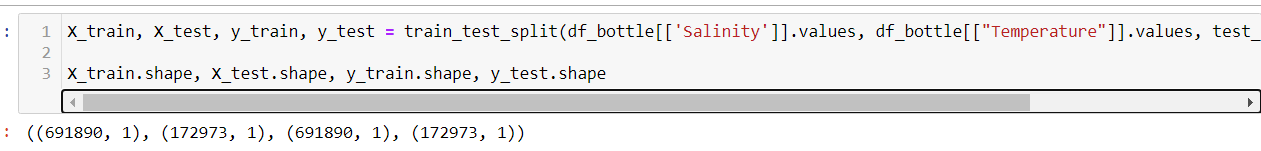


Рисунок 15 - Разделение на обучающую и тестовую выборки

## Линейная регрессия

Для того, чтобы решить задачу влияния солёности на температуру воды, необходимо использовать различные модели регрессии. Начнем с одной из самых простых моделей, модели линейной регрессии. Линейная регрессия является моделью зависимости независимой переменной x и зависимой (выходной) переменной y.

Сделаем обучение модели линейной регрессии по обучающей выборке, предскажем результат температуры, а также, создадим датафрейм (См. Рисунок 16). Созданный датафрейм будет состоять из нескольких столбцов с данными:

1. Соленость воды – Salinity
2. Температура воды – Temperature (Реальное значение температуры воды)
3. Температура воды предсказанное – Temp\_pred\_linear\_salinity (Предсказанное с помощью модели линейной регрессии температуры воды)

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 16 - Обучение модели линейной регрессии

В ходе выполнения курсовой работы, после каждого обучения какой-либо модели, будет создан датафрейм, включающий в себя вышеперечисленные столбцы, было принято решение не пояснять создание остальных датафреймов с целью уменьшения дублирования информации. В следующей рассматриваемой модели будут включены сразу 2 признака в вектор признаков x, такие как соленость воды и глубина воды. Соответственно, будет создан похожий на (См. Рисунок 16) датафрейм, состоящий из таких же столбцов, но с добавлением нового столбца (глубина воды).

Построим график сравнения предсказанной температуры воды от истинной температуры. Предсказанная температура отмечена зеленым цветом, а истинная синим.

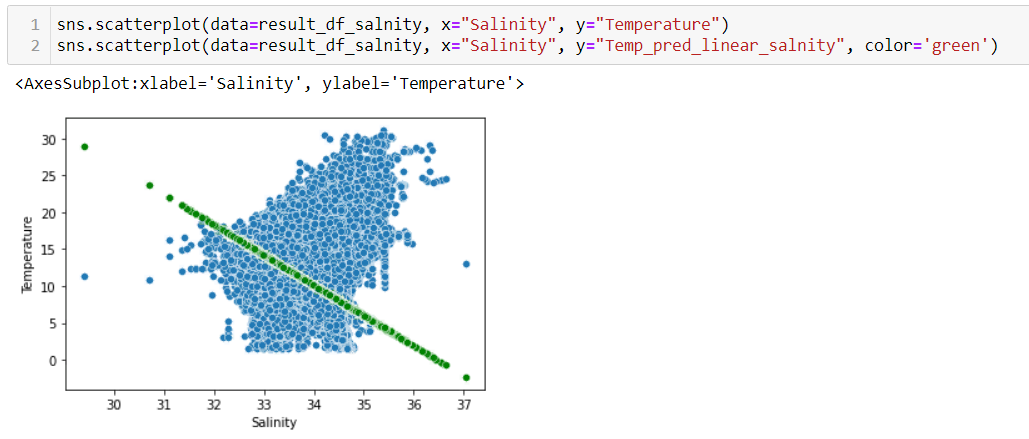


Рисунок 17 - График сравнения

Для того, чтобы оценить эффективность модели линейной регрессии, а также точность и качество предсказания, выведем оценки для прогнозируемых данных (См. Рисунок 18).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 18 – Оценки для прогнозируемых данных

Как мы видим с помощью графика и оценок, модель линейной регрессии не точная и не эффективная, так как коэффициент детерминации слишком маленький, 21%, соответственно, точность верного прогноза достаточно низкая. Ошибки прогноза (MAE, MSE) самые высокие по сравнению с другими моделями. Чем больше данные показатели, тем хуже рассматриваемая модель.

## Множественная регрессия

Так как результат прогноза линейной регрессии оказался достаточно плохим, мы можем сделать вывод о том, что прогнозировать результат только по одной входной переменной x недостаточно, включим в набор входных значений x данные по столбцам солености и глубины воды. Моделью зависимости нескольких независимых входных переменных x и зависимой выходной переменной y, называется множественная регрессия.

Разделим набор данных на обучающую и тестовую выборки в соотношении 80/20, включив в вектор признаков x данные по столбцам солености и глубины воды, данные по столбцу температуры передадим в качестве целевого вектора y (См. Рисунок 19).

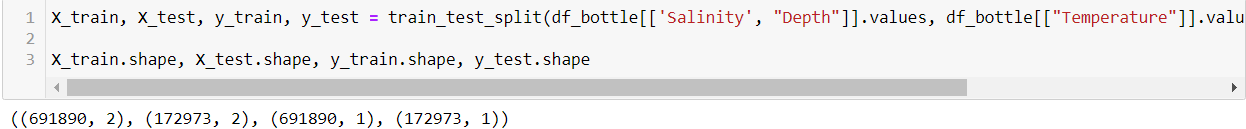


Рисунок 19 - Разделение на обучающую и тестовые выборки

Обучаем модель множественной регрессии по обучающей выборке, а также, предскажем результат температуры (См. Рисунок 20).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 20 - Обучение модели множественной регрессии

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 21). Предсказанная температура отмечена оранжевым цветом, а истинная синим.

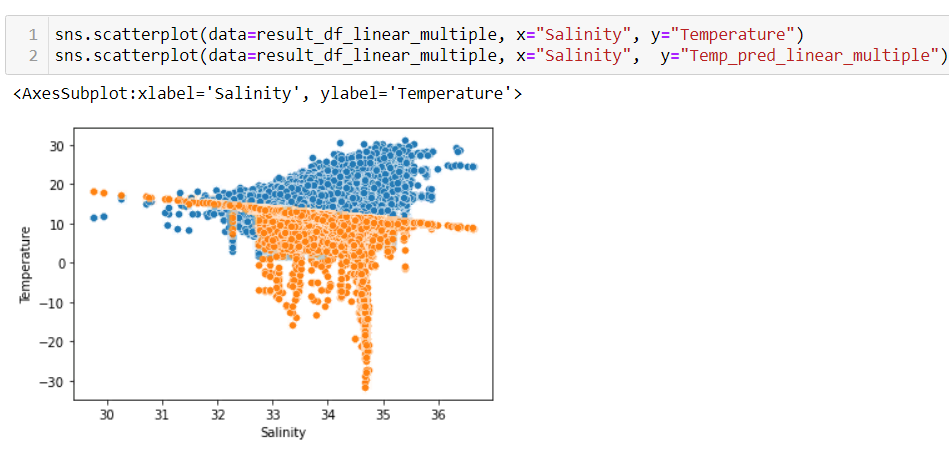


Рисунок 21 - График сравнения

Выводим оценки для прогнозируемых данных (См. Рисунок 22).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 22 – Оценки прогнозируемых данных

Как мы можем заметить, модель множественной регрессии оказалась эффективнее и точнее, чем модель линейной регрессии, так как коэффициент детерминации стал выше: 47%, ошибки прогноза (MSE, MAE) уменьшились. Исходя из данного вывода, обучать модель по нескольким независимым переменным будет эффективнее, все последующие модели будут включать в себя несколько входных независимых переменных и одну выходную зависимую переменную.

## Полиномиальная регрессия

Рассмотрим следующую модель регрессии – полиномиальную регрессию. Полиномиальная регрессия является моделью зависимости независимых переменных x и зависимой переменной у, моделируется как полином заданной n степени в x. Такой алгоритм используется для того, чтобы обучить линейную модель на нелинейных данных, а также способствует существенному улучшению результатов прогнозирования за счет добавления дополнительных функций к данным.

Создадим функцию для ввода заданной пользователем степени полиномиальной регрессии, в данной функции произойдет преобразование исходного набора данных при помощи полиномиальной регрессии, после этого проводится обучение модели по преобразованной обучающей выборке, с предсказанием результата температуры. Оценки прогнозируемых данных и коэффициенты модели включены в данную функцию (См. Рисунок 23).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 23 - Функция для работы с моделью полиномиальной регрессии

Вызываем вышеописанную функцию для полиномиальной регрессии, со степенью полинома равной 2 (См. Рисунок 24).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 24 - Вызов функции для 2 степени полинома

Можно увидеть то, что полиномиальная регрессия оказалась эффективнее первых двух моделей, так как коэффициент детерминации стал значительно выше: 66%, а также ошибки прогноза (MAE, MSE) уменьшились.

График сравнения предсказанной от истинной температуры воды (См. Рисунок 26).

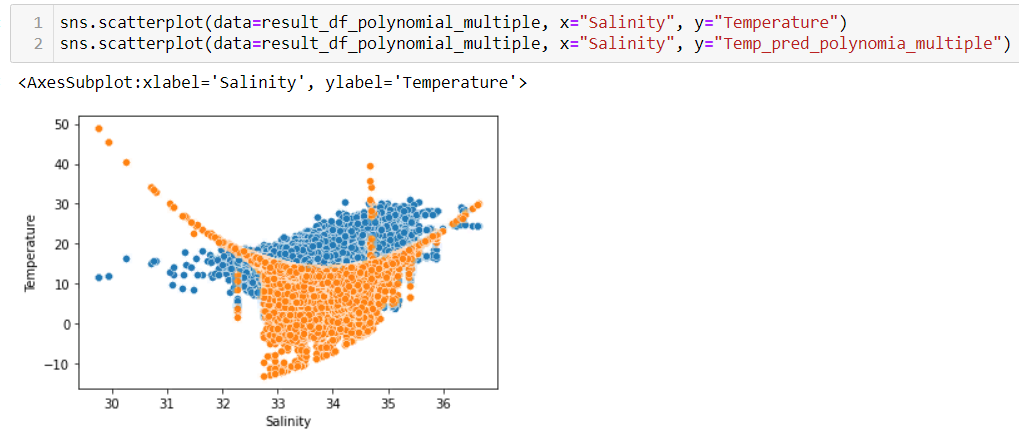


Рисунок 25 - График сравнения

Построим полиномиальную регрессию со степенью полинома равной 3 (См. Рисунок 26).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 26 - Вызов функции для 3 степени полинома

Результат прогноза стал еще лучше. Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 27).

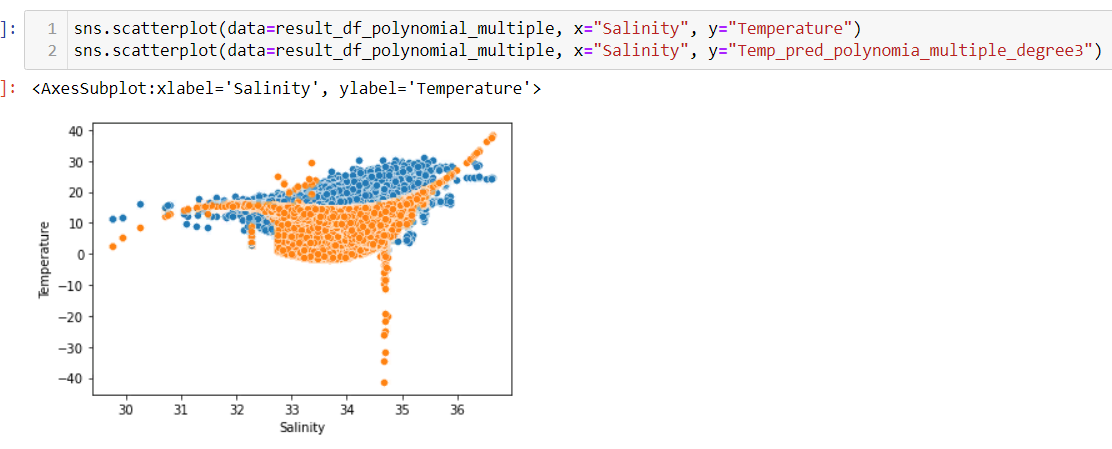


Рисунок 26 - График сравнения

Изменим степень полинома до 5 и посмотрим на результат прогноза (См. Рисунок 27), а также выведем график сравнения предсказанной от истинной температуры воды (См. Рисунок 28).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 27 - Вызов функции для 5 степени полинома

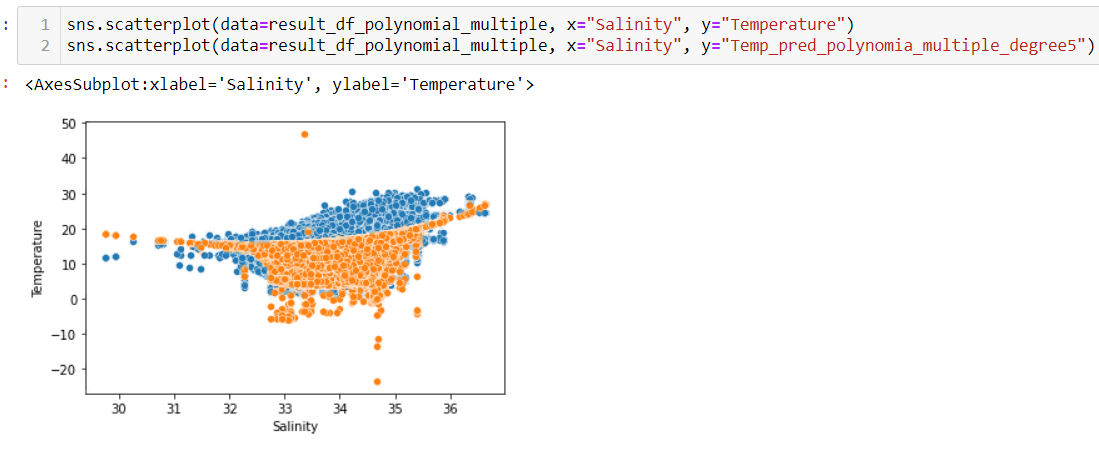


Рисунок 28 - График сравнения

На основании оценок прогноза, самой эффективной моделью среди полиномиальных регрессий является полиномиальная регрессия с 5 степенью полинома, так как коэффициент детерминации стал равен 78%, точность верного прогноза достаточно высокая, средняя абсолютная (MAE) и средняя квадратическая ошибки стали меньше. Можем сделать вывод о точности и высоком качестве предсказания результата температуры у данной модели.

## Случайный лес

Рассмотрим модель случайного леса. Это модель, которая состоит из нескольких деревьев решений, каждое из которых строится на случайной выборке из исходного набора данных, также в каждом узле дерева выбирается случайным образом подмножество объектов с целью создания оптимального, или же наилучшего разделения. Схема построения деревьев соответствует принципу ансамблирования. В нашем случае ансамблирование — это когда алгоритм машинного обучения строится на базе нескольких решающих деревьев.

Создадим функцию для ввода заданного пользователем количества деревьев в случайном лесу. В данной функции произойдет обучение модели по обучающей выборке, с предсказанием результата температуры. Оценки прогнозируемых данных включены в данную функцию (См. Рисунок 29).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 29 - Функция для работы с моделью случайного леса

Вызовем данную функцию для случайного леса из 10 деревьев (См. Рисунок 30).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 30 - Вызов функции для 10 деревьев

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 31).

Изображение выглядит как карта

Автоматически созданное описание

Рисунок 31 - График сравнения

Как мы можем заметить, исходя из оценок и графика, на данный момент случайный лес, по сравнению с остальными моделями является самой точной моделью:

1. Коэффициент детерминации значительно повысился: 80%
2. Средняя квадратическая ошибка (MSE) стала меньше: 3.5744
3. Средняя абсолютная ошибка (MAE) стала меньше: 1.2036

Сделаем вывод, что данная модель является эффективной и хорошей. Точность верного прогноза высокая, а ошибки прогноза низкие. Известно, что при увеличении количества деревьев, точность и качество модели увеличивается.

Построим модель случайного леса с увеличением количества деревьев до 50 (См. Рисунок 32).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 32 - Случайный лес из 50 деревьев

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 33).

Изображение выглядит как карта

Автоматически созданное описание

Рисунок 33 - График сравнения

Увеличение количества деревьев до 50 незначительно улучшило модель:

1. Коэффициент детерминации совсем немного увеличился: с 80.47% до 80.59%
2. Средняя Абсолютная ошибка немного уменьшилась: c 1.1982 до 1.1944
3. Средняя квадратичная ошибка немного уменьшилась: c 3.54 до 3.52
4. Время обучения модели увеличилось почти в 5 раз: с 12 до 59,7 секунд

Построим модель случайного леса с увеличением количества деревьев до 200 (См. Рисунок 34).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 34 - Случайный лес из 200 деревьев

График сравнения предсказанной от истинной температуры воды (См. Рисунок 35).

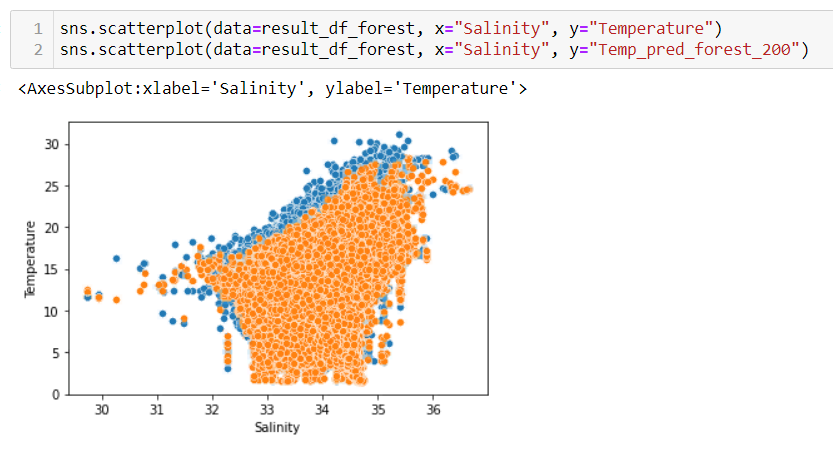


Рисунок 35 - График сравнения

Увеличение количества деревьев до 200 незначительно улучшило модель случайного леса:

1. Коэффициент детерминации чуть-чуть увеличился: с 80.59% до 80.65%
2. Средняя Абсолютная ошибка немного уменьшилась: c 1.1944 до 1.1927
3. Средняя квадратичная ошибка немного уменьшилась: c 3.52 до 3.50
4. Время обучения модели увеличилось почти в 4 раза: с 1 до 4 минуты

На основании оценок прогноза, самой эффективной моделью является модель случайного леса из 200 деревьев. Отметим, что при увеличении количества деревьев, модель случайного леса будет более точная и качественная, но время обучения при каждом увеличении будет сильно увеличиваться.

## Деревья решений

Рассмотрим модель деревьев решений. Деревья решений являются базовой единицей рассмотренного нами алгоритма случайного леса, а также позволяют получить высокую точность в ходе решения задач. Идея состоит в том, чтобы создать модель, предсказывающую значения целевой переменной у с помощью правил принятия решений. Одной из главных особенностей данной модели является четкость представления информации. Стоит отметить, что деревья решений представляют правила, которые находятся в последовательной структуре, где каждому объекту соответствует узел, в котором находится и выдается решение. Данная модель может обрабатывать, как и числовые, так и категориальные данные.

Обучаем модель деревьев решений по обучающей выборке и предсказываем результат температуры (См. Рисунок 36).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 36 - Обучение и предсказание результата

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 37).

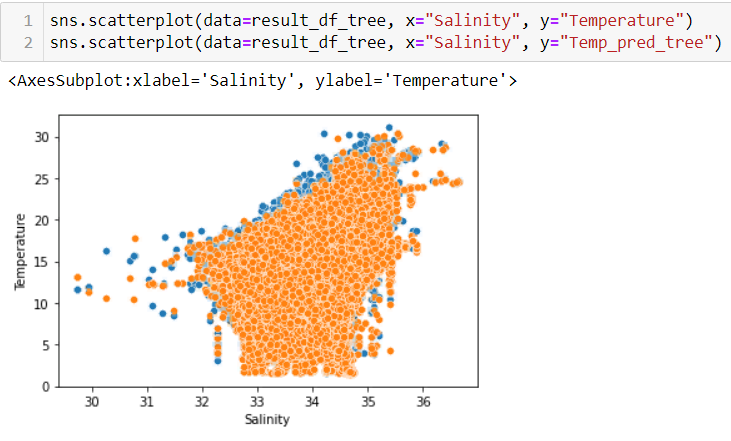


Рисунок 37 - График сравнения

Выведем оценки для прогнозируемых данных (См. Рисунок 38).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 38 - Оценки для прогнозируемых данных

На основании оценок прогноза модель дерева решений является достаточно хорошей моделью. Сравним ее с самой точной и эффективный на данный момент моделью – моделью случайного леса из 200 деревьев:

1. Коэффициент детерминации у модели дерева решений = 78%, а у случайного леса = 80.65%, сделаем вывод о том, что точность верного прогноза выше у случайного леса
2. Средняя абсолютная ошибка у модели дерева решений = 1.2527, а у случайного леса = 1.1927, ошибка прогноза у случайного леса меньше
3. Средняя квадратическая ошибка у модели дерева решений = 3.953, а у случайного леса = 3.50, ошибка прогноза меньше у модели случайного леса.

Сделаем вывод о том, в данном случае модель дерева решений является менее эффективной и точной, чем модель случайного леса.

## Регрессия LASSO

Рассмотрим модель лассо регрессии. Лассо регрессия – это модель, оценивающая разреженные коэффициенты, позволяющая уменьшить ограничения модели, адаптированная специально для данных, демонстрирующих сильную мультиколлинеарность. Для получения более устойчивого решения вводится дополнительное слагаемое регуляризации в функционал оптимизации модели. Регуляризация – это техника, позволяющая сократить данные до нулевого значения. Стоит отметить, что Лассо использует сжатие коэффициентов. Сжатие коэффициентов – процесс приближения данных к центральной точке.

Обучаем модель лассо регрессии по обучающей выборке, предсказываем результат температуры воды (См. Рисунок 39).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 39 - Обучение и предсказание результата

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 40).

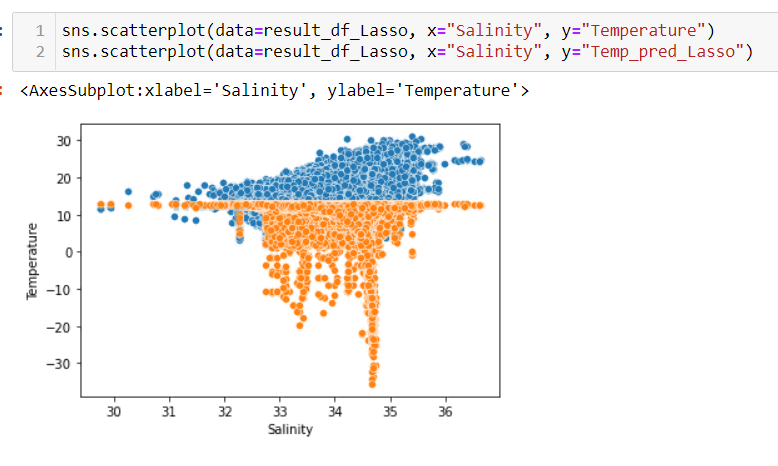


Рисунок 40 - График сравнения

Выведем оценки для прогнозируемых данных (См. Рисунок 41).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 41 - Оценки для прогнозируемых данных

На основании графика сравнения и оценок прогноза можно сделать вывод о неэффективности и низкой точности Лассо регрессии, так как коэффициент детерминации достаточно низкий: 45.5%, ошибки прогноза (Средняя абсолютная и квадратичная ошибка) высокие.

## Гребневая регрессия

Рассмотрим модель гребневой регрессии. Гребневую регрессию можно отнести к регуляризованной версии линейной регрессии. Данная модель похожа на модель Лассо регрессии, рассмотренную выше, потому что она тоже применяет сжатие коэффициентов. Различие между этими двумя моделями в том, что гребневая регрессия применяет регуляризацию L2, в которой ни один из коэффициентов не становится нулевым. Считается, что гребневая регрессия подходит для ситуаций, в которых следует учитывать большое количество переменных, имеющих небольшой эффект, в то время как Лассо регрессия учитывает меньшее количество переменных, имеющих большой или средний эффект.

Обучим модель гребневой регрессии на обучающем наборе данных, предскажем результат температуры воды (См. Рисунок 42).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 42 - Обучение и предсказание результата

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 43).

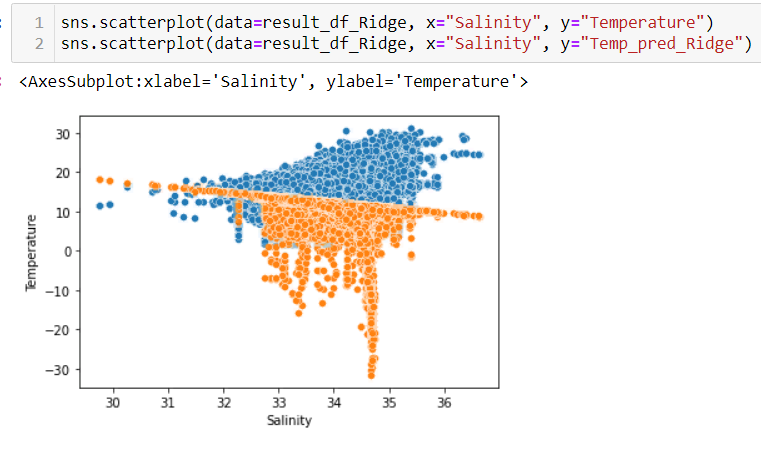


Рисунок 43 - График сравнения

Выведем оценки для прогнозируемых данных (См. Рисунок 44).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 44 - Оценки для прогнозируемых данных

На основании графика сравнения, а также оценок прогнозируемых данных можно сделать вывод о неэффективности и достаточно низкой точности модели гребневой регрессии. Коэффициент детерминации слишком низкий: 47%, что говорит нам о низкой точности верного прогноза, ошибки прогноза достаточно высокие: MAE = 2.3319, MSE = 9.6076.

## Метод k ближайших соседей

Рассмотрим модель ближайших соседей. Данная модель используется в случае, когда метки данных непрерывные. Метка вычисляется на основе среднего значения меток ее ближайших соседей и присваивается точке запроса. Увеличение количества соседей будет иметь тенденцию сглаживать границы решений.

Обучим модель ближайших соседей по 2 соседям, предскажем результат температуры воды (См. Рисунок 45).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 45 - Обучение и предсказание результата

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 46).

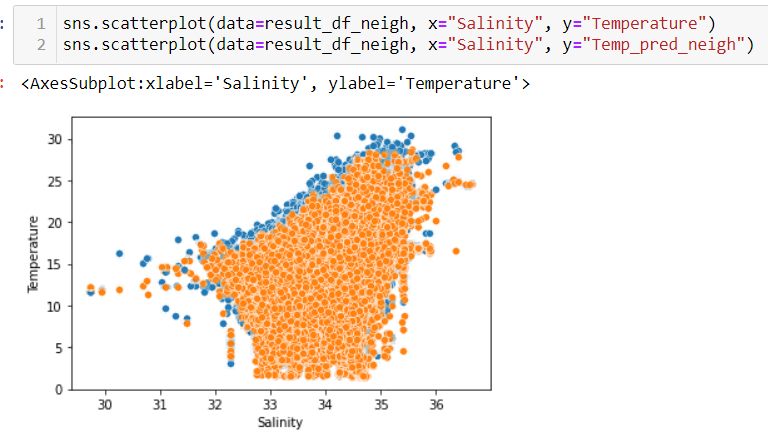


Рисунок 46 - График сравнения

Выводим оценки для прогнозируемых данных (См. Рисунок 47).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 47 - Оценки для прогнозируемых данных

Как мы можем заметить, коэффициент детерминации у данной модели средний, а время обучения составляет не более 10 секунд, средняя абсолютная и квадратическая ошибки не особо высокие. Попробуем обучить модель k ближайших соседей на разных количествах соседей.

Обучим модель ближайших соседей по 3 соседям, предскажем результаты температуры воды (См. Рисунок 48).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 48 - Обучение и предсказание результата

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 49).

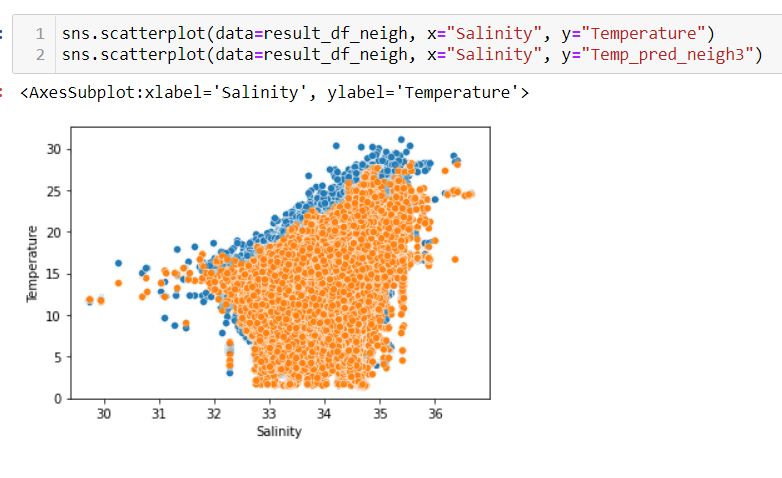


Рисунок 49 - График сравнения

Выводим оценки для прогнозируемых данных (См. Рисунок 50).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 50 - Оценки для прогнозируемых данных

Мы видим, что при увеличении количества соседей до 3, коэффициент детерминации увеличился, время обучения модели увеличилось всего на 1 секунду, средняя абсолютная и квадратическая ошибки уменьшились.

Обучим модель ближайших соседей по 7 соседям, а также сделаем предсказание результата температуры воды (См. Рисунок 51).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 51 - Обучение и предсказание результата

Построим график сравнения предсказанной от истинной температуры воды (См. Рисунок 52).

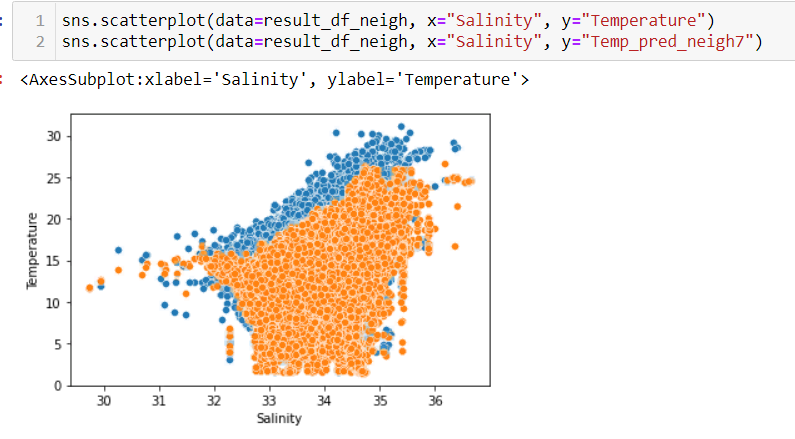


Рисунок 52 - График сравнения

Выводим оценки для прогнозируемых данных (См. Рисунок 53).

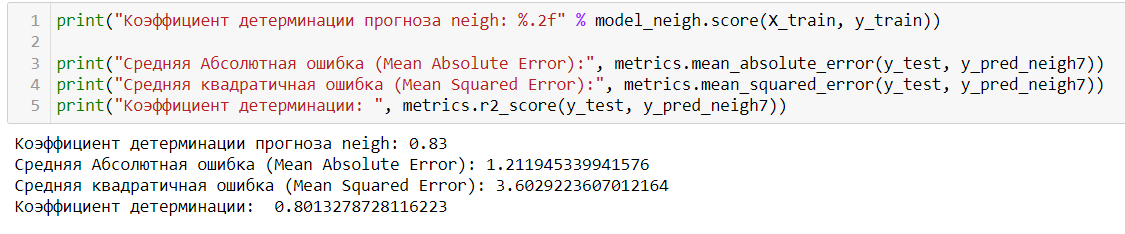


Рисунок 53 - Оценки для прогнозируемых данных

Модель стала еще более эффективная, коэффициент детерминации значительно увеличился, в то время как время обучение почти не изменилось, оно так же составляет 11 секунд, ошибки прогноза уменьшились.

Обучим модель ближайших соседей по 10 соседям, предскажем результат температуры воды (См. Рисунок 54).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 54 - Обучение и предсказание результата

Построим график сравнения предсказанной от истинной температуры (См. Рисунок 55).

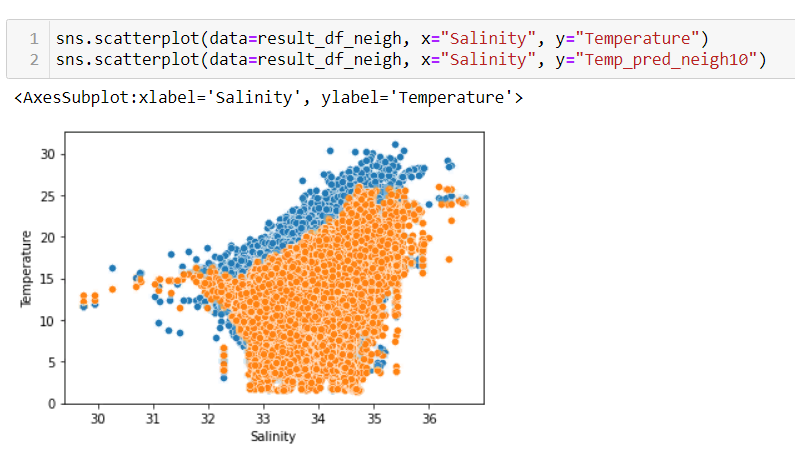


Рисунок 55 - График сравнения

Выводим оценки для прогнозируемых данных (См. Рисунок 56).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 56 - Оценки для прогнозируемых данных

На основании оценок прогнозируемых данных можно сделать вывод о том, что с увеличением количества соседей коэффициент детерминации будет увеличиваться, а ошибки прогноза уменьшаться. Модель k ближайших соседей является эффективной и точной моделью.

# Выбор модели

Попробуем найти по результатам метрик самую точную модель. Находить и выбирать мы будем на основании оценок прогнозируемых данных по каждой модели (метрике).

1. Коэффициент детерминации: у модели k ближайших соседей самое высокое значение данной метрики, равное 80.78%
2. Средняя квадратическая ошибка: у модели k ближайших соседей самое низкое значение данной метрики, равное 3.4854
3. Средняя абсолютная ошибка: у модели k ближайших соседей самое низкое значение данной метрики, равное 1.1915

По результатам всех метрик, результаты модели k ближайших соседей лучше, чем у всех остальных моделей. Сделаем вывод о том, что выбором самой точной и эффективной моделью будет модель k ближайших соседей. Второй по точности и эффективности является модель случайного леса.

## Grid Search

Grid search, или же, поиск по сетке, является алгоритмом машинного обучения, использующимся для повышения точности модели машинного обучения путем отбора лучших гиперпараметров. Поиск по сетке принимает на вход модель и ее гиперпараметры. Данный метод считает ошибку для каждого сочетания значений гиперпараметров, выбирает такое сочетание, в котором ошибка минимальна. Гиперпараметрами называют переменные модели, которые позволяют управлять процессом обучения модели. В данной работе мы уже сталкивались с гиперпараметрами модели:

1. При работе с полиномиальной регрессией - передавали гиперпараметр количества степеней полинома «degree» в модель (См. Рисунок 26).
2. При работе со случайным лесом – когда передавали количество деревев «n\_estimators» в модель (См. Рисунок 32).
3. При работе с моделью ближайших соседей – когда передавали количество соседей «n\_neighbors» в модель (См. Рисунок 45).

Как было сказано ранее, самой точной моделью является модель ближайших соседей, а второй по точности является модель случайного леса. Попробуем применить к ним поиск по сетке для того, чтобы их усовершенствовать.

Используем Grid Search для модели случайного леса. В нашем случае, при увеличении количества деревьев росло время обучения модели, в то время как точность увеличивалась незначительно. С помощью поиска по сетке попробуем добиться компромиссом между желаемой точностью и временем обучения модели. Создадим сетку гиперпараметров, а также экземпляр GridSearch (См. Рисунок 57). В данном случае мы будет подбирать количество деревьев (n\_estimators) и максимальную глубину дерева (max\_depth).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 57 - Создание и настройка GridSearch

Обучение GridSearch по сетке гиперпараметров (См. Рисунок 58).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 58 - Обучение GridSearch

После обучения получим лучшие гиперпараметры (См. Рисунок 59).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 59 - Лучшие гиперпараметры

После получения лучших гиперпараметров построим модель случайного леса, обучим ее на обучающем наборе данных, предскажем результат температуры воды (См. Рисунок 60).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 60 - Обучение и предсказание по лучшим гиперпараметрам

Выводим оценки для прогнозируемых данных (См. Рисунок 61).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 61 - Оценки для прогнозируемых данных

На основании оценок сделаем вывод о том, что поиск по сетке никак не улучшил точность модели. Попробуем ввести другие значения гиперпараметров в сетку. (См. Рисунок 62).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 62 - Создание и настройка GridSearch

Обучение GridSearch по сетке гиперпараметров (См. Рисунок 63).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 63 - Обучение GridSearch

После обучения получим лучшие гиперпараметры (См. Рисунок 64).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 64 - Лучшие гиперпараметры

Поиск по сетке выделил гиперпараметры max\_depth = 7, n\_estimators = 20 лучшими. Построим модель случайного леса, внесем в нее значения лучших гиперпараметров, обучим модель случайного леса на обучающем наборе данных, предскажем результат температуры воды (См. Рисунок 65).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 65 – Создание модели, обучение, предсказание результата

Выведем оценки для прогнозируемых данных (См. Рисунок 66).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 66 - Оценки для прогнозируемых данных

Поиск по сетке улучшил точность модели случайного леса:

1. Коэффициент детерминации до поиска был равен 80.65%, а после стал равен 81.49%
2. Средняя абсолютная ошибка до поиска была равна 1.1927, а после стала равна 1.1855
3. Средняя квадратическая ошибка до поиска была равна 3.5085, а после стала равна 3.356
4. Время обучения до поиска была равна более 4 минут, а после стала равна 9 секунд

Коэффициент детерминации увеличился, ошибки MAE, MSE снизились, время обучения существенно снизилось. Сделаем вывод о том поиск по сетке (GridSearch) значительно улучшил модель случайного леса.

Используем Grid Search для модели k ближайших соседей. Создадим сетку гиперпараметров, а также экземпляр GridSearch (См. Рисунок 67). В данном случае мы будет подбирать количество соседей (n\_neighbors), весовую функцию, используемую при прогнозировании (weights) и параметр мощности для метрики Минковского (p).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 67 - Создание и настройка GridSearch

Обучение GridSearch по сетке гиперпараметров (См. Рисунок 68).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 68 - Обучение GridSearch

После обучения получим лучшие гиперпараметры (См. Рисунок 69).

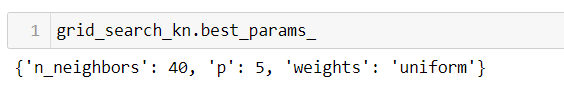


Рисунок 69 - Лучшие гиперпараметры

Поиск по сетке выделил гиперпараметры n\_neighbors = 7, p = 5, weights = ‘uniform’ лучшими. Построим модель ближайших соседей, внесем в нее значения лучших гиперпараметров, обучим модель ближайших соседей на обучающем наборе данных, предскажем результат температуры воды (См. Рисунок 70).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 70 - Создание модели, обучение, предсказание результата

Выводим оценки для прогнозируемых данных (См. Рисунок 71).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 71 - Оценки для прогнозируемых данных

Поиск по сетке улучшил точность модели ближайших соседей:

1. Коэффициент детерминации до поиска был равен 80.78%, а после стал равен 81.53%
2. Средняя абсолютная ошибка до поиска была равна 1.1915, а после стала равна 1.1688
3. Средняя квадратическая ошибка до поиска была равна 3.4854, а после стала равна 3.3348
4. Время обучения после поиска стала немного медленнее, равна 15 секундам

Коэффициент детерминации увеличился, ошибки MAE, MSE снизились, время обучения незначительно повысилась примерно на 4 секунды. Сделаем вывод о том поиск по сетке (GridSearch) значительно улучшил модель k ближайших соседей.

# Анализ полученных результатов

В данном разделе приведем в виде таблицы основные результаты проделанной работы. Каждая строка соответствует исследуемому алгоритму, каждый столбец – значению одной из его метрик.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Коэффициент детерминации, % | Средняя абсолютная ошибка | Средняя квадратичная ошибка |
| Линейная регрессия | 21 | 2,82 | 14,47 |
| Множественная регрессия | 47 | 2,33 | 9,61 |
| Полиномиальная регрессия степени 5 | 78 | 1,41 | 4,03 |
| Случайный лес | 81,49 | 1,19 | 3,36 |
| Деревья решений | 78,2 | 1,25 | 3,95 |
| Регрессия LASSO | 45,5 | 2,43 | 9,88 |
| Гребневая регрессия | 47,02 | 2,33 | 9,61 |
| Метод k-ближайших соседей | 81,54 | 1,16 | 3,34 |

Таблица 1. Результаты работы моделей

Приведем некоторые комментарии к таблице 1. В ней указаны данные для моделей с лучшими гиперпараметрами. Такие модели, как случайный лес и метод k-ближайших соседей имею гиперпараметры, полученные с помощью алгоритма поиска по сетке.

# Заключение

В результате проделанной работы было произведено сравнение методов регрессии на реальном наборе данных. Для решения была выбрана задача прогнозирования температуры воды Мирового океана по данным о её солености. Для этого мы рассмотрели и сравнили 8 основных моделей регрессии. Кроме того, с целью получения оптимального набора гиперпараметров моделей, был использован алгоритм Grid Search, реализующий поиск по сетке гиперпараметров. В ходе его работы была повышена точность моделей случайного леса и k ближайших соседей.

Был сделан вывод о том, что соленость воды действительно влияет на температуру воды, с увеличением количества солености температура воды увеличивалась. Наилучший результат показала модель k ближайших соседей с количеством соседей, равным 40. Для этого алгоритма коэффициент детерминации равен 81,53%, средняя абсолютная ошибка равна 1,17, средняя квадратическая ошибка равна 3,33.

# Ссылка на репозиторий с полным кодом выполнения работы

<https://github.com/Karambasss/Course_work_ML_salinity_temp>

# Список использованных источников

1. Geron A. Hand on Machine Learning with scikit-learn and Tensorflow. - O'Reilly Media, 2017. - 564 с.
2. Albon C. Machine learning with Python Handbook. - O'Reilly Media, 2018. - 427 с.
3. Coelho L.P., Richert W. Building machine learning systems with Python. - Packt Publishing, 2015. - 326 с.
4. Grus J. Data science from scratch. - O'Reilly Media, 2015. - 330 с.
5. Machine Learning in Python // scikit-learn URL: https://scikit-learn.org/stable/index.html (дата обращения: 18.04.2022).
6. Как проверить качество модели с помощью метрик // Nplus1 URL: https://nplus1.ru/material/2020/03/27/course-data-science-chapter-6 (дата обращения: 20.04.2022).