ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ М.В.ЛОМОНОСОВА»

ЭКОНОМИЧЕСКИЙ ФАКУЛЬТЕТ

«УТВЕРЖДАЮ»

Декан экономического факультета МГУ
профессор. _____ А.А.Аузан

«» …….. 2020 год

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Машинное обучение и анализ данных-1

Уровень высшего образования
Магистратура

Направление подготовки (специальность)
38.04.01 Экономика

Направленность (профиль) ОПОП
Анализ данных в экономике

Форма обучения
очная

Рабочая программа рассмотрена и одобрена
Учебно-методической комиссией экономического факультета
(протокол №_____, дата)

Москва 2020

На оборотной стороне титульного листа указывается:

Рабочая программа дисциплины разработана в соответствии с самостоятельно установленным МГУ образовательным стандартом (ОС МГУ) для реализуемых основных профессиональных образовательных программ высшего образования по направлению подготовки «_____» магистратуры

ОС МГУ утвержден решением Ученого совета МГУ имени М.В.Ломоносова от _____20 _____года (протокол №__).

Год (годы) приема на обучение: 2020 и последующие

1. **Место и статус дисциплины в структуре основной профессиональной образовательной программы подготовки магистра**

Статус дисциплины: *вариативная*
Триместр: *2*

2. **Входные требования для освоения дисциплины**

Для успешного освоения данного курса требуются знания и навыки, полученные в следующих дисциплинах:

— теория вероятностей и математическая статистика;
— основы программирования.

3. **Планируемые результаты обучения по дисциплине, соотнесенные с требуемыми компетенциями выпускников**

Формируемые компетенции	**Планируемые результаты обучения по дисциплине, соотнесенные с требуемыми компетенциями**
Способность формулировать научно обоснованные гипотезы, создавать теоретические модели явлений и процессов, применять методологию научного познания в профессиональной деятельности **(М.УК-1)**	**УМЕТЬ** выдвигать научно обоснованные гипотезы, поддающиеся операционализации, моделировать явления и процессы на основе системного видения различных отраслей знаний **М.УК-1.Ум.1**
Способность применять продвинутые инструментальные методы экономического анализа в прикладных и/или фундаментальных исследованиях **(М.ОПК-5)**	**УМЕТЬ** обрабатывать информацию при помощи методов машинного обучения **М.ОПК-5.Ум.1**
Способность проводить самостоятельные исследования в соответствии с разработанной программой **(М.ПК-3)**	**ЗНАТЬ** современные научные методы машинного обучения и анализа данных **М.ПК-3.Зн.1**
	УМЕТЬ применять современные научные методы машинного обучения и анализа данных в экономических исследованиях **М.ПК-3.Ум.1**
Способность представлять результаты проведенного исследования научному сообществу в виде статьи или доклада **(М.ПК-4)**	**УМЕТЬ** представлять результаты научного исследования в систематизированном виде в письменной форме **М.ПК-4.Ум.1**
	УМЕТЬ создавать презентации по итогам исследований и делать устные научные доклады **М.ПК-4.Ум.2**

Способность анализировать и использовать различные источники информации для проведения экономических расчетов **(М.ПК-9)**	**УМЕТЬ** оценивать качество источников экономической информации **М.ПК-9.Ум.1**
	УМЕТЬ применять качественные и количественные методы для проведения прикладных экономических исследований **М.ПК-9.Ум.2**
Способность разрабатывать эконометрические модели и модели машинного обучения исследуемых экономических процессов и явлений, интерпретировать полученные результаты **(М.СПК-1)**	**ЗНАТЬ** современные инструментальные методы, применяемые в экономических исследованиях **М.СПК-1.Зн.1**
	УМЕТЬ применять современные инструментальные методы к релевантным данным для решения заданного или самостоятельно сформулированного исследовательского вопроса **М.СПК-1.Ум.1**
	УМЕТЬ представлять результаты аналитических расчетов, полученные с применением современных инструментальных методов **М.СПК-1.Ум.2**
Способность видеть логические связи в системе собранной, обработанной и проанализированной информации, и на основании этого разрабатывать рекомендации для лиц, принимающих решения на микро- и макроуровне, или бизнес-решения **(М-СПК-4)**	**УМЕТЬ** при помощи методов анализа данных сравнивать альтернативные решения и находить оптимальные по заданным метрикам качества **М.СПК-4.Ум.1**
	УМЕТЬ на основе сделанных выводов об оптимальности решения разрабатывать рекомендации для лиц, принимающих решения **М.СПК-4.Ум.2**

4. **Объем дисциплины по видам занятий**

Объем дисциплины составляет 6 зачетных единицы: 216 академических часов, из которых 56 академических часов составляет аудиторная нагрузка, из них 28 академических часов — лекции и 28 академических часов — семинары, 52 академических часа — групповая контактная работа, 0 академических часов — индивидуальная контактная работа, 108 академических часов составляет самостоятельная работа магистранта.

5. **Формат обучения:** используется электронная информационная среды экономического факультета МГУ имени М.В.Ломоносова «ON.ECON».

6. **Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий**

Название раздела/темы	Всего, часы	В том числе			
		Контактная работа с преподавателем			Самостоятельная работа магистранта, часы
		Лекции, часы	Семинары, часы	Групповая, часы	
Тема 1. Основные задачи машинного обучения и анализа данных		2	2	4	0
Тема 2: Основные библиотеки для работы с данными в Python		2	2	4	9
Тема 3. Линейные модели в задаче регрессии		2	2	4	9
Тема 4. Линейные модели в задаче классификации		2	2	4	9
Тема 5. Оценки качества моделей в задачах регрессии и классификации		2	2	4	9
Тема 6. Выбор модели. Кросс-валидация. Отбор признаков		2	2	4	9
Тема 7. Регуляризация. Преобразование признаков		2	2	4	9
Тема 8. Метод опорных векторов		2	2	4	9
Тема 9. Деревья решений		2	2	4	9
Тема 10. Ансамбли моделей		2	2	4	9
Тема 11. Введение в нейронные сети		4	4	4	9
Тема 12. Анализ временных рядов		4	4	8	18
Текущая аттестация: *— домашние задания*					
Всего	**216**	**28**	**28**	**52**	

Краткое содержание тем дисциплины

Тема 1. Основные задачи машинного обучения и анализа данных. Типы задач: обучение с учителем (регрессия, классификация), обучение без учителя (кластеризация, поиск аномалий, снижение размерностей), частичное обучение, обучение с подкреплением. Основные области: компьютерное зрение, обработка естественного языка, рекомендательные системы, анализ временных рядов, обучение ранжированию, построение выводов по данным. Типы данных. Библиотеки научных вычислений: NumPy, SciPy. Библиотека для работы с табличными данными Pandas. Библиотеки визуализации: Matplotlib, Seaborn. Среда интерактивных вычислений Jupyter Notebook: настройка и установка, основные принципы работы.

Тема 2: Регрессия и классификация. Математическая постановка задач регрессии и классификации. Метрические методы регрессии и классификации: метод ближайших соседей, взвешенный метод ближайших соседей. Расстояния в пространстве признаков. Генеративная модель: идеальный байесовский классификатор, наивный байесовский классификатор.

Тема 3. Линейные модели в задаче регрессии. Метод наименьших квадратов, метод максимального правдоподобия, свойства оценок параметров модели. Аналитическое решение, итерационные методы обучения. Обобщения: взвешенный МНК, локальная регрессия.

Тема 4. Линейные модели в задаче классификации. Логистическая регрессия для бинарной классификации: метод максимального правдоподобия, метод наименьших квадратов с итеративным пересчетом весов, градиентный спуск. Многоклассовая классификация: один против всех, Softmax.

Тема 5. Оценки качества моделей в задачах регрессии и классификации. Метрики в задачи регрессии: MAE, MSE, MAPE, R^2. Метрики в задаче классификации: кросс-энтропия, precision, recall, F-мера, ROC-кривая, AUC ROC. Информационные метрики: AIC, BIC, SBC.

Тема 6. Выбор модели. Кросс-валидация. Отбор признаков. Обобщающая способность и ее оценка: отложенная выборка, кросс-валидация. Отбор признаков в линейных моделях: stepwise-регрессия.

Тема 7. Регуляризация. Преобразование признаков. Проблема переобучения. L_1- и L_2-регуляризация. Гребневая регрессия, LASSO-регрессия, регрессия наименьшего угла, ElasticNet. Методы обучения моделей с регуляризацией. Виды признаков: категориальные, вещественные.

Тема 8. Метод опорных векторов. Метод опорных векторов в задаче классификации. Ядерный переход. Метод опорных векторов для задачи регрессии.

Тема 9. Деревья решений. Деревья решений для задачи классификации. Алгоритмы построения деревьев: ID3, C4.5, CART. Стрижка дерева. Связь с линейными моделями.

Тема 10. Ансамбли моделей. Bootstrap-метод. Бэггинг. Стэкинг. Случайный лес. Дилемма смещения-дисперсии. Бустинг: Adaboost, градиентный бустинг.

Тема 11. Введение в нейронные сети. История нейронных сетей. Искусственный нейрон. Многослойная нейронная сеть в задачах регрессии и классификации. Дифференцирование

функции, заданной графом. Метод обратного распространения ошибок. Методы обучения нейронных сетей, основанные на стохастическом градиентном спуске.

Тема 12. Анализ временных рядов. Временные ряды. Стационарность временных рядов. Модели авторегрессии–скользящего среднего (модели ARMA, ARIMA, SARIMA, SARIMAX). Гетероскедастичность временных рядов (модель GARCH). Временные ряды с трендом (модель Хольта–Уинтерса).

Фонд оценочных средств для оценивания результатов обучения по дисциплине

Шкала оценивания результатов (баллы) по дисциплине:

Результаты обучения по дисциплине	Виды оценочных средств
УМЕТЬ выдвигать научно обоснованные гипотезы, поддающиеся операционализации, моделировать явления и процессы на основе системного видения различных отраслей знаний **М.УК-1.Ум.1**	Домашние работы Выполнение проектных заданий
УМЕТЬ обрабатывать информацию при помощи методов машинного обучения **М.ОПК-5.Ум.1**	Домашние работы Выполнение проектных заданий
ЗНАТЬ современные научные методы машинного обучения и анализа данных **М.ПК-3.Зн.1**	Домашние работы Выполнение проектных заданий
УМЕТЬ применять современные научные методы машинного обучения и анализа данных в экономических исследованиях **М.ПК-3.Ум.1**	Домашние работы Выполнение проектных заданий
УМЕТЬ представлять результаты научного исследования в систематизированном виде в письменной форме **М.ПК-4.Ум.1**	Домашние работы Выполнение проектных заданий
УМЕТЬ создавать презентации по итогам исследований и делать устные научные доклады **М.ПК-4.Ум.2**	Домашние работы Выполнение проектных заданий
УМЕТЬ оценивать качество источников экономической информации **М.ПК-9.Ум.1**	Домашние работы Выполнение проектных заданий
УМЕТЬ применять качественные и количественные методы для проведения прикладных экономических исследований **М.ПК-9.Ум.2**	Домашние работы Выполнение проектных заданий

УМЕТЬ выдвигать научно обоснованные гипотезы, поддающиеся операционализации, моделировать явления и процессы на основе системного видения различных отраслей знаний **М.УК-1.Ум.1**	Домашние работы Выполнение проектных заданий
УМЕТЬ обрабатывать информацию при помощи методов машинного обучения **М.ОПК-5.Ум.1**	Домашние работы Выполнение проектных заданий
ЗНАТЬ современные научные методы машинного обучения и анализа данных **М.ПК-3.Зн.1**	Домашние работы Выполнение проектных заданий
УМЕТЬ при помощи методов анализа данных сравнивать альтернативные решения и находить оптимальные по заданным метрикам качества **М.СПК-4.Ум.1**	Домашние работы Выполнение проектных заданий
УМЕТЬ на основе сделанных выводов об оптимальности решения разрабатывать рекомендации для лиц, принимающих решения **М.СПК-4.Ум.2**	Домашние работы Выполнение проектных заданий

Виды оценочных средств	**Баллы**
Домашние работы	200
Проектные задания	100

Оценка по дисциплине выставляется, исходя из следующих критериев:

Оценка	**Минимальное количество баллов**	**Максимальное количество баллов**
Отлично	255	300
Хорошо	195	254,9
Удовлетворительно	120	194,9
Неудовлетворительно	60	119,9

Примечание: в случае, если магистрант за триместр набирает менее 20% баллов от максимального количества по дисциплине, то уже на промежуточном контроле (и далее на пересдачах) действует следующее правило сдачи: «магистрант может получить только оценку «Удовлетворительно», и только если получит за промежуточный контроль, включающий весь материал дисциплины, не менее, чем 85% от баллов за промежуточный контроль».

Типовые задания, методические рекомендации по их подготовке и требования к их выполнению:

Домашние работы представляют собой практические задания, ориентированные на закрепление навыков машинного обучения и анализа данных на языке Python в рамках различных тем. Задание сдается в формате Jupyter-ноутбука .ipynb с кодом, комментариями и ответами на вопросы задания. В комментариях следует описать используемые для решения задачи методы и данные, подробно обосновать выбор алгоритма, представить результаты расчётов (при необходимости используя таблицы и рисунки). Домашние задания позволяют набрать до 200 баллов.

Проектные работы представляют собой практические задания, требующие от студента применения полученных навыков машинного обучения для решения проблем. Задание сдается в формате устной презентации, а также исходного кода на языке Python в удобном формате (например, Jupyter-ноутбук, код на репозитории Github, zip-архив). Проект может быть как индивидуально, так и в команде. В ходе защиты проекта требуется кратно описать решаемую задачу, описать применяемые методы машинного обучения и анализа данных, сформулировать и объяснить полученные результаты, ответить на вопросы принимающих. Проектные работы позволяют набрать до 100 баллов.

Ниже приведены примеры заданий, которые будут задаваться студентам в процессе обучения.

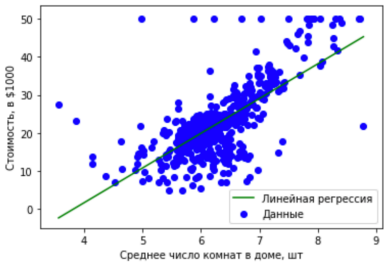
Пример 1: Реализация линейной регрессии для предсказания цен на недвижимость.

Обучим нашу модель предсказывать среднюю цену квартиры по среднему числу комнат в доме.

In [5]:
```
predictors = ['rm']

# формируем данные для модели
# метод .values возвращает данные в виде np.array
X = boston[predictors].values
y = boston['medv'].values

model_lm1 = LinearRegression()
model_lm1.fit(X, y)
y_pred = model_lm1.predict(X)
```

Теперь посмотрим как выглядит предсказание обученной модели.

In [6]:
```
# сделаем сетку от минимального до максимального значения среднего числа комнат
X_low, X_high = X.min(), X.max()
X_plt = np.linspace(X_low, X_high, 64)
y_pred_plt = model_lm1.predict(X_plt.reshape(-1, 1))

# график предсказаний и диаграмма рассеяния
plt.plot(X_plt, y_pred_plt, color='green', label='Линейная регрессия')
plt.scatter(X[:, 0], y, color='blue', label='Данные')

plt.xlabel('Среднее число комнат в доме, шт')
plt.ylabel('Стоимость, в $1000')
plt.legend()
plt.show()
```



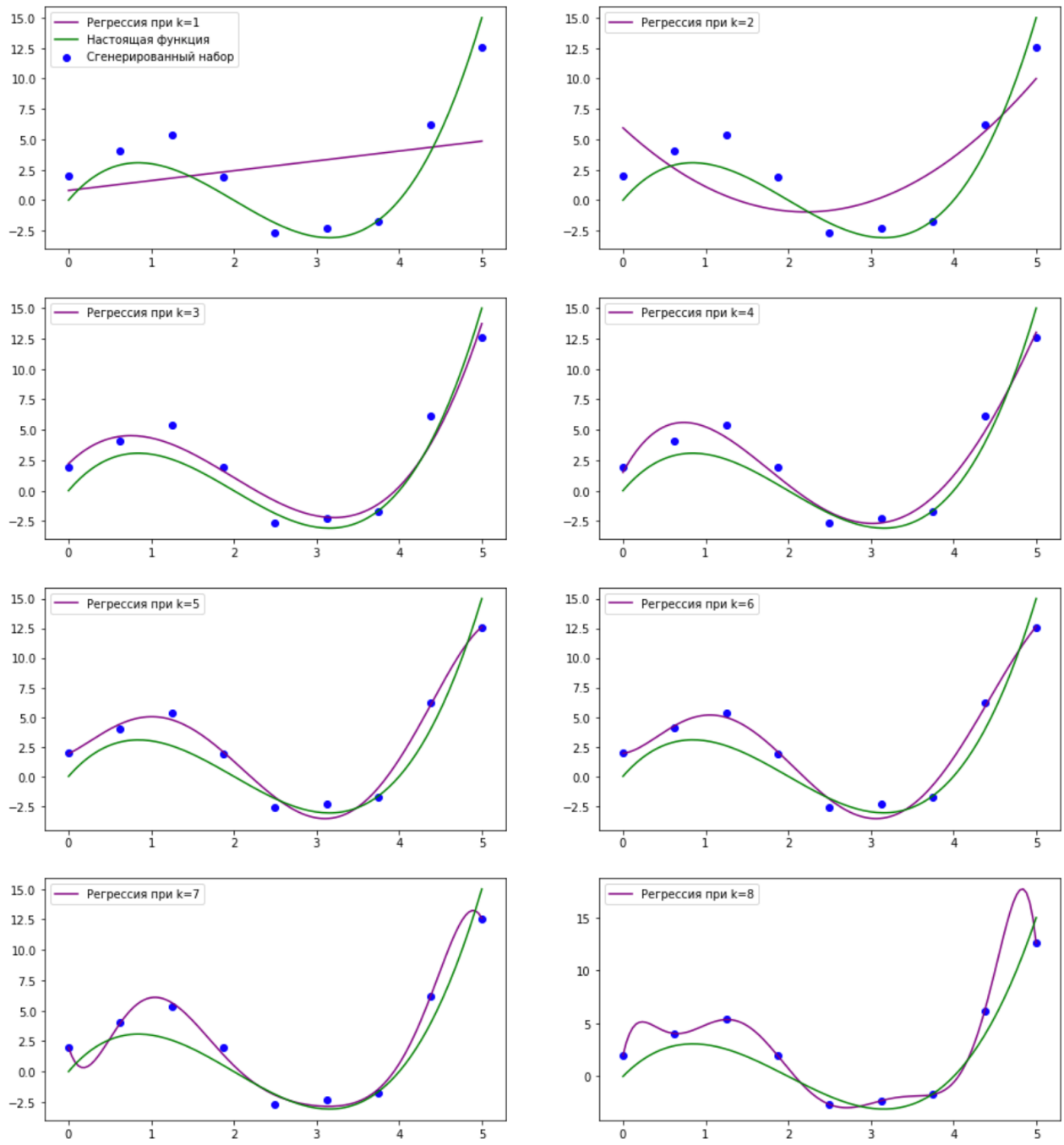
Пример 2: Проблема переобучения в задаче полиномиальной регрессии (с решением).

Задание №3 (2 балла): Определить функцию, которая строит матрицу признаков:

In [20]:
```
# на вход передается вектор x размера N
# на выходе выдать матрицу Xp размера Nxk,
# где i-ая строка соответствует степеням x_i
# с 1 до k включительно (нулевая степень будет добавлена в методе fit)
def get_polynomial_features(X, k):
    ### МЕСТО ДЛЯ ВАШЕГО КОДА ###

    # РЕШЕНИЕ
    X = np.array(X)
    return np.column_stack([X ** i for i in range(1, k+1)])
    # КОНЕЦ РЕШЕНИЯ
```

Обучим модель полиномиальной регрессии и посмотрим на график предсказаний для различных k.

In [21]:
```
# настоящая зависимость
x_plt = np.linspace(0, 5, 128)
y_plt = func(x_plt)

_, axes = plt.subplots(4, 2, figsize=(16, 18))
for k, ax in zip(range(1, 9), axes.flatten()):
    mdl = LinearRegression()
    Xp = get_polynomial_features(X, k)
    mdl.fit(Xp, y)
    yp_plt = mdl.predict(get_polynomial_features(x_plt, k))
    ax.plot(x_plt, yp_plt, color='purple', label=f'Регрессия при k={k}')
    ax.plot(x_plt, y_plt, color='green', label='Настоящая функция' if k == 1 else None)
    ax.scatter(X, y, color='blue', label='Сгенерированный набор' if k == 1 else None)
    ax.legend()
plt.show()
```



Пример 3: реализация алгоритма гребневой регрессии на языке Python (с решением).

- *L2-регуляризация (Ridge-регрессия или гребневая регрессия)*. К функции потерь добавляется L_2 норма весов θ:

$$\theta^{rigde} = \arg\min_{\theta \in \Theta} \left\{ \sum_{k=1}^{n} S(y_k, f_\theta(x_k)) + \alpha \sum_{i=1}^{d} \theta_i^2 \right\},$$

где $\alpha > 0$ -- гиперпараметр. Аналитическое решение задачи оптимизации в случае L2-регуляризации:

$$\theta^{ridge} = (X^T X + \alpha I)^{-1} X^T y.$$

Задание №4 (12 баллов): применить гребневую регрессию на наборе данных по недвижимости в Бостоне и сравнить результаты с обычной линейной регрессией.

Требуется:

- Реализовать метод **fit** настройки параметров (6 баллов).
- Реализовать метод **predict**, который предсказывает целевую переменную (4 балла).
- Сравнить метрики гребневой регрессии и обычной линейной регрессии (2 балла).

Замечание: Для того, чтобы гребневая регрессия работала корректно, необходимо привести все признаки к единым величинам. Для этого для каждого признака x из него вычитают выборочное среднее и делят на корень из выборочной дисперсии:

$$\widetilde{x} = \frac{x - \text{mean}(x)}{\text{std}(x)}, \quad \text{mean}(x) = \frac{1}{n}\sum_{k=1}^{n} x_k, \quad \text{std}(x) = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(x - \text{mean}(x))^2}$$

In [22]:
```
# Реализация гребневой регрессии линейной регрессии
class RidgeRegression:
    # настройка модели
    def __init__(self, alpha):
        self.alpha = alpha
        self._is_fitted = False
        return

    # метод fit
    # на вход принимает матрицу X размера Nxd с объясняющими признаками
    # и вектор y размера N с предсказанными значениями целевой переменной
    # настраивает веса модели
    # возвращает сам себя (для совместимости с библиотекой sklearn)
    def fit(self, X, y):
        # добавляем фиктивный признак
        X = np.array(X)
        y = np.array(y)

        # выполним нормализацию признака
        # важно сделать это ДО добавления фиктивного признака
        # иначе он превратится в 0 после нормализации
        self._mean = X.mean(axis=0, keepdims=True)
        self._std = X.std(axis=0, keepdims=True)

        X = (X - self._mean) / self._std

        # дописываем фиктивный признак
        X = np.column_stack([np.ones(len(X)), X])

        # настроить параметры theta
        # self._theta = ### МЕСТО ДЛЯ ВАШЕГО КОДА ###

        # РЕШЕНИЕ
        d = X.shape[1]
        I = np.eye(d)
        self._theta = np.linalg.inv(X.T @ X + self.alpha * I) @ X.T @ y
        # КОНЕЦ РЕШЕНИЯ

        self._is_fitted = True
        return self

    # на вход принимает матрицу X размера Nxd с объясняющими признаками
    # на выход дает вектор y размера N с предсказанными значениями целевой переменной
    # примечание: не забудьте нормализовать признаки
    def predict(self, X):
        ### МЕСТО ДЛЯ ВАШЕГО КОДА ###

        # РЕШЕНИЕ
        X = (np.array(X) - self._mean) / self._std

        X = np.column_stack([np.ones(len(X)), X])
        return X @ self._theta
        # КОНЕЦ РЕШЕНИЯ
```

8. Ресурсное обеспечение

8.1.Перечень основной и дополнительной литературы

Основная литература:

- Christopher M. Bishop. Pattern Recognition and MachineLearning
- T.Hastie, R. Tibshirani, J. Friedman. Elements of statisticallearning
- Kevin P. Murphy. Machine Learning: A Probablistic Perspective
- Петер Флах. Машинное обучение. Наука и искусствопостроения алгоритмов, которые извлекают знания изданных
- В.Н.Вапник. Восстановление зависимостей поэмпирическим данным.

- В.Н.Вапник, А.Я.Червоненкис. Теория распознаванияобразов.
- Vladimir N. Vapnik. The Nature of Statistical Learning Theory.

8.2.Перечень лицензионного программного обеспечения

- Среда разработки Anaconda для языка Python.
- Библиотеки анализа данных для языка Python.

8.3.Перечень профессиональных баз данных и информационных справочных систем

- Школа Анализа Данных Яндекса
- Академия MADE
- Coursera

8.4.Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

- Курс CS229 Stanford
- Видеолекции курса CS299 Stanford
- Towards Data Science
- Kaggle
- Блог А.Г. Дьяконова
- Курс Open Data Science

8.5. Описание материально-технической базы

Для организации занятий по дисциплине необходимы следующие технические средства обучения: компьютерный класс с проектором.

Для организации дистанционных занятий по дисциплине необходимы следующие технические средства: компьютер с доступом в интернет, камера и микрофон, аккаунт Zoom и установленная среда разработки Anaconda у каждого студента.

1. **Язык преподавания:** русский.

2. **Преподаватель (преподаватели):** Гончаренко В.В.

3. **Автор (авторы) программы:** Гончаренко В.В.