Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»
Кафедра информатики

Лабораторная работа №3

«Переобучение и регуляризация.»

Выполнил: Чёрный Родион Павлович
магистрант кафедры информатики
группа №858642

Проверил: доцент кафедры информатики
Стержанов Максим Валерьевич

Минск 2019

Постановка задачи

Набор данных **ex3data1.mat** представляет собой файл формата *.mat. Набор содержит две переменные X (изменения уровня воды) и y (объем воды, вытекающий из дамбы). По переменной X необходимо предсказать y. Данные разделены на три выборки: обучающая выборка, по которой определяются параметры модели; валидационная выборка, на которой настраивается коэффициент регуляризации; контрольная выборка, на которой оценивается качество построенной модели.

1. Загрузите данные **ex3data1.mat** из файла.
2. Постройте график, где по осям откладываются X и y из обучающей выборки.
3. Реализуйте функцию стоимости потерь для линейной регрессии с L2-регуляризацией.
4. Реализуйте функцию градиентного спуска для линейной регрессии с L2-регуляризацией.
5. Постройте модель линейной регрессии с коэффициентом регуляризации 0 и постройте график полученной функции совместно с графиком из пункта 2. Почему регуляризация в данном случае не сработает?
6. Постройте график процесса обучения для обучающей и валидационной выборки. По оси абсцисс откладывается число элементов из обучающей выборки, а по оси ординат - ошибка для обучающей выборки и валидационной выборки. Какой вывод можно сделать по построенному графику?
7. Реализуйте функцию добавления p - 1 новых признаков в обучающую выборку (X^2, X^3, X^4, …, X^p).
8. Поскольку в данной задаче будет использован полином высокой степени, то необходимо перед обучением произвести нормализацию признаков.
9. Обучите модель с коэффициентом регуляризации 0 и p = 8.
10. Постройте график модели, совмещенный с обучающей выборкой, а также график процесса обучения. Какой вывод можно сделать в данном случае?
11. Постройте графики из пункта 10 для моделей с коэффициентами регуляризации 1 и 100. Какие выводы можно сделать?

12. С помощью валидационной выборки подберите коэффиент регуляризации, который позволяет достичь наименьшей ошибки. Процесс подбора отразите с помощью графика (графиков).
13. Вычислите ошибку (потерю) на контрольной выборке.

Описание реализации

1. Загрузим из файла ex3data1.mat данные для обучения, валидации и тестирования модели. Сразу же проведем нормализацию данных.

```
mat = scipy.io.loadmat('ex3data1.mat')
X = mat['X'] / np.linalg.norm(mat['X'])
y = mat['y'] / np.linalg.norm(mat['y'])
X_val = mat['Xval'] / np.linalg.norm(mat['Xval'])
y_val = mat['yval'] / np.linalg.norm(mat['yval'])
X_test = mat['Xtest'] / np.linalg.norm(mat['Xtest'])
y_test = mat['ytest'] / np.linalg.norm(mat['ytest'])
```

2. На рисунке 1 изображен график, где по отложены точки из обучающей выборки.

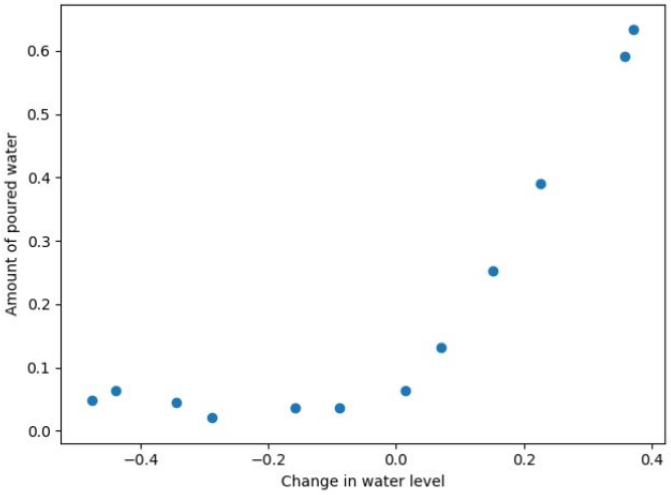

Рисунок 1 - Данные об изменении уровня воды и количестве вытекаемой воды из обучающей выборки

3. Реализуем функцию стоимости для линейной регрессии с учетом коэффициента регуляризации:

```
def mse(self):
    error = np.dot(self.X, self.model_parameters.T) - self.Y
    reg = self.reg_coeff * self.model_parameters.dot(self.model_parameters)
    return (np.dot(error, error.T) + reg) / (2 * self.n)
```

4. Реализуем функцию градиентного спуска для линейной регрессии с учетом коэффициента регуляризации:

```
def calculate_descent_direction(self):
    E = np.dot(self.X, self.model_parameters.T) - self.Y
    dw0 = np.dot(E, self.X.T[0]) / self.n
    dw = [dw0]
    for i in range(1, self.features):
        dwi = np.dot(E, self.X.T[i])
        dw.append((dwi + self.reg_coeff * self.model_parameters[i]) / self.n)
    dw = np.array(dw)
    return dw
```

5. Произведем 40 итераций градиентного спуска для модели линейной регрессии с шагом обучения 1. Изображение полученного графика вместе с исходными данными приведено на рисунке 2.

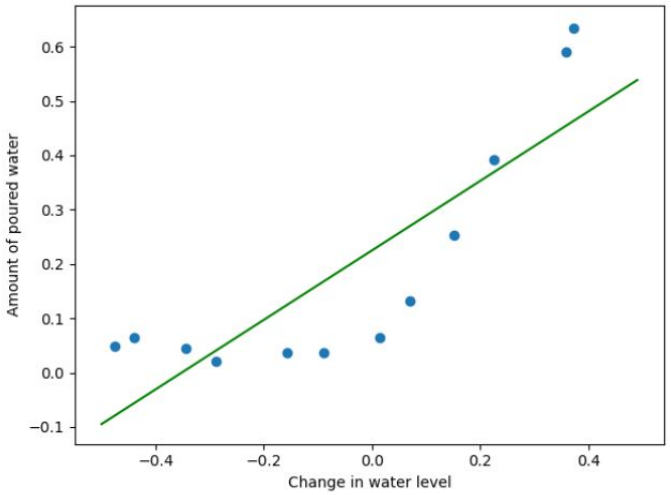

Рисунок 2 - Модель линейной регрессии для исходных данных

6. Обучим модель на валидационной выборке, состоящей из 21 элемента. Сравнение кривых обучения обоих случаев приведено на рисунке 3. Как видно из сравнения, валидационная выборка позволила обучить модель на более точное предсказание значений.

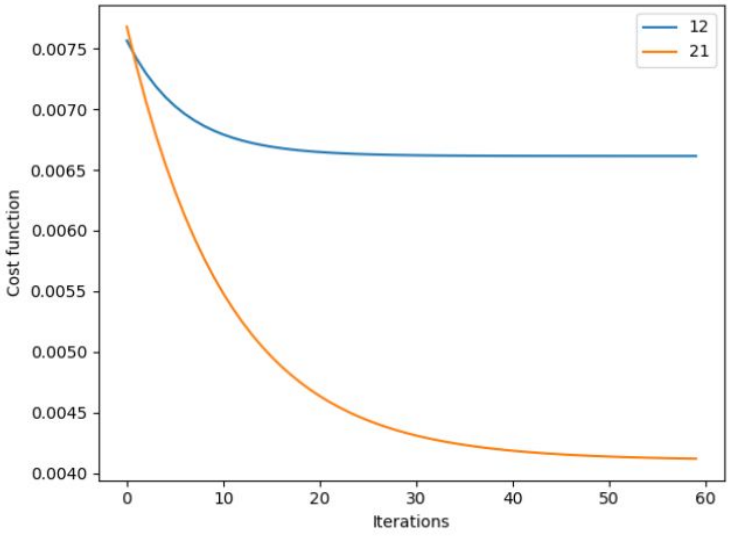

Рисунок 3 - Кривые обучения модели, обученной на тренировочной выборке (синяя кривая) и на валидационной (оранжевая)

7. Для добавления новых признаков в обучающую выборку воспользуемся классом PolynominalFeatures:

```
poly = PolynomialFeatures(degree=p)
self.X = poly.fit_transform(x)
```

8. Нормализация данных уже была проведена в пункте 1.

9. Обучим модель с коэффициентом регуляризации 0 и степенью полинома 8:

```
reg = LinearRegression(X, Y, p=8, learning_rate=1, reg_coeff=0)
reg.train(n_iterations=500000)
```

10. График полученной модели вместе с исходными данными изображен на рисунке 4. Из полученного графика можно сделать вывод, что добавленные полиномиальные признаки позволяют нам строить кривую, которая, при достаточной тренировке, сможет пройти через все точки и достигнуть нулевого значения функции стоимости.

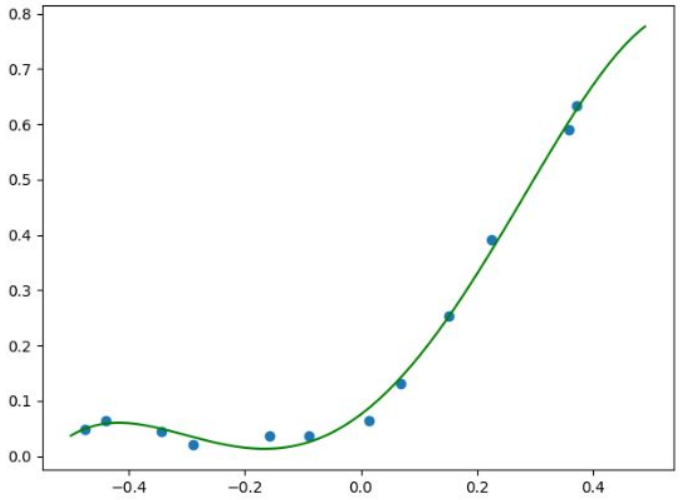

Рисунок 5 - График модели, полученный при степени полинома 8 и 500000 итерациях градиентного спуска

11. Повторим ту же процедуру, но используя коэффициенты регуляризации 1 и 100. Сравнение результатов приведено на рисунке 6. Глядя на него можно сделать вывод, что данные коэффициенты регуляризации слишком упрощают полиномиальную модель, сводя ее к линейному случаю.

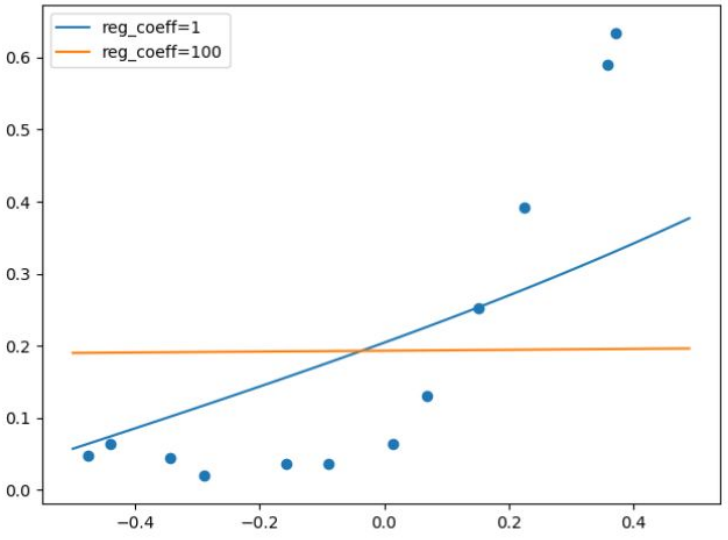

Рисунок 6 - Графики моделей, полученные при коэффициентах регуляризации 1 и 100

12. Используем валидационную выборку, чтобы подобрать оптимальный коэффициент ругляризации, который бы сократил вероятность переобучения, но при этом сохранил бы точность модели. Будем перебирать коэффициенты в небольшом диапазоне (подобранном эмпирически): от 0.0001 до 0.0009.

Сравнение полученных кривых приведено на рисунке 7, а сравнение точности этих моделей на валидационной выборке - на рисунке 8.

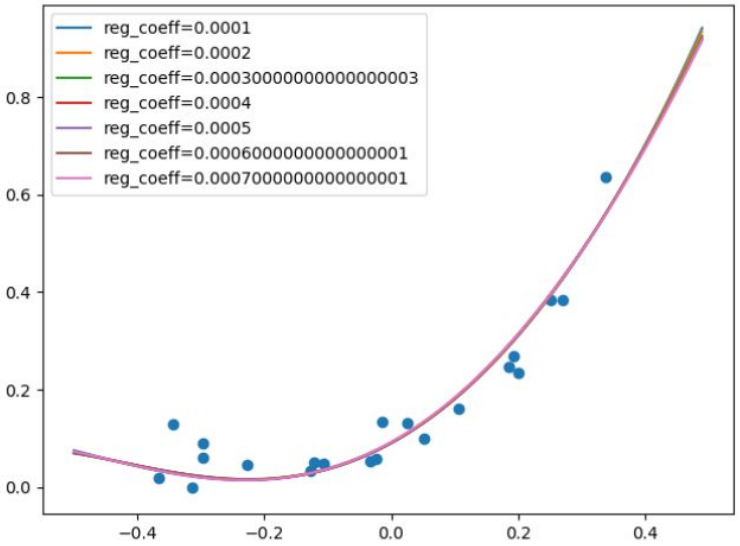

Рисунок 7 - Сравнение графиков моделей для различных коэффициентов регуляризации

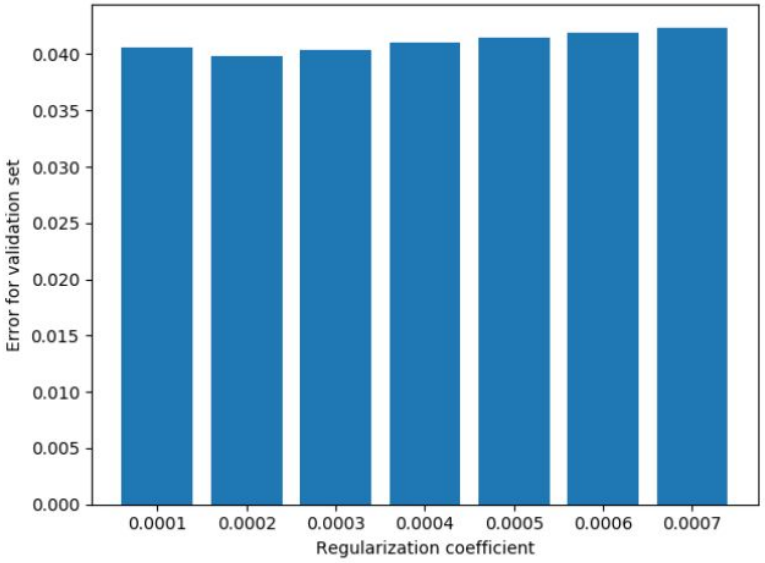

Рисунок 8 - Значения ошибок на валидационной выборке для различных параметров регуляризации

13. Из рисунка 8 видно, что наибольшая точность достигается при коэффициенте регуляризации 0.0002. Это и будет коэффициентом регуляризации для нашей финальной модели. Ошибка на контрольной выборке, при этом, составляет 0.0324.

Выводы

Линейная модель, при достаточно высокой степени полинома и достаточной обученности, может выводить нелинейную зависимость между входными данными. Однако тут возникает проблема переобучения, когда модель показывает очень точный результат на обучающих данных, но, при этом, плохо предсказывает всё остальное.

Для устранения проблемы переобучения можно использовать L2 регуляризацию. Однако, если значение параметра регуляризации будет слишком высоким, то мы получим недообученную модель. Поэтому, необходимо подбирать оптимальное значение этого параметра на валидационном наборе данных.