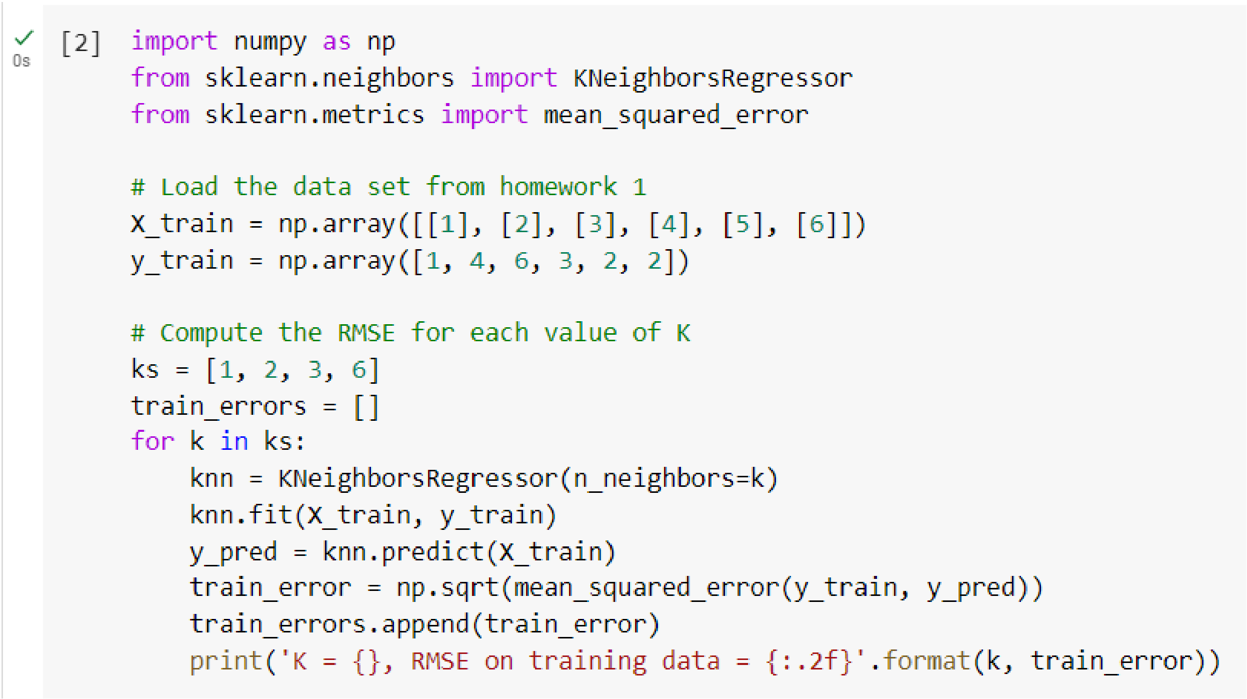
Homework 2

Name: Sahiti Gabrani | UID: 123456789

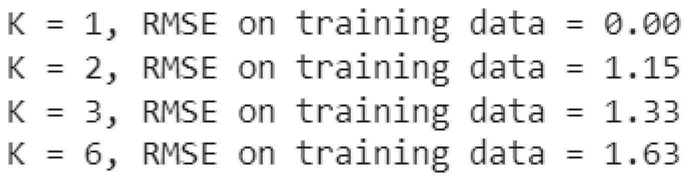
ECE M148 | Homework 2
Introduction to Data Science | Due: April 19, 12:00 p.m.
Instructor: Lara Dolecek | TAs: Harish G V, Jayanth Shreekuma

Q1.

root mean squared error (RMSE)) on the training data in homework 1 for each choice of K for K = 1, 2, 3, and 6:



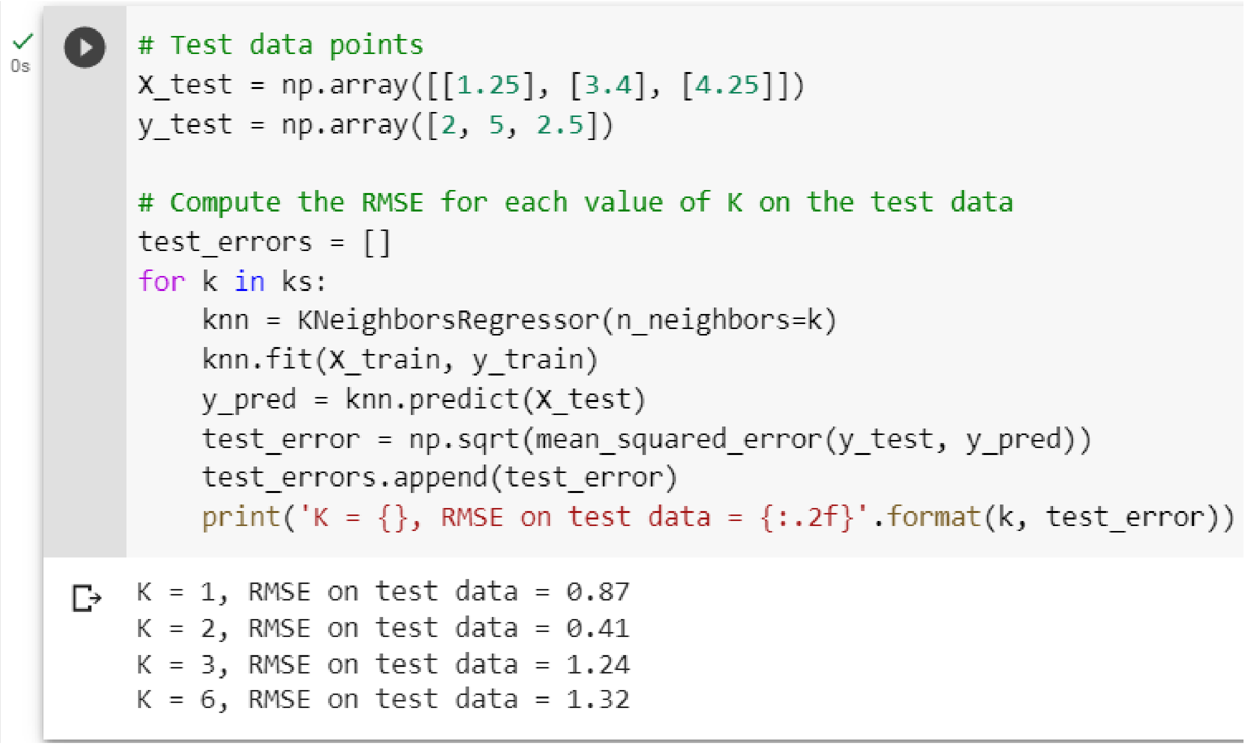
```python
import numpy as np
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import mean_squared_error

# Load the data set from homework 1
X_train = np.array([[1], [2], [3], [4], [5], [6]])
y_train = np.array([1, 4, 6, 3, 2, 2])

# Compute the RMSE for each value of K
ks = [1, 2, 3, 6]
train_errors = []
for k in ks:
    knn = KNeighborsRegressor(n_neighbors=k)
    knn.fit(X_train, y_train)
    y_pred = knn.predict(X_train)
    train_error = np.sqrt(mean_squared_error(y_train, y_pred))
    train_errors.append(train_error)
    print('K = {}, RMSE on training data = {:.2f}'.format(k, train_error))
```



```
K = 1, RMSE on training data = 0.00
K = 2, RMSE on training data = 1.15
K = 3, RMSE on training data = 1.33
K = 6, RMSE on training data = 1.63
```

As we can see, the KNN regressor with K=1 has an RMSE of 0 on the training data, indicating perfect fit to the training data. We will choose the value of K = 1 for the dataset.

Test RSME and regression on the points in A:



```
# Test data points
X_test = np.array([[1.25], [3.4], [4.25]])
y_test = np.array([2, 5, 2.5])

# Compute the RMSE for each value of K on the test data
test_errors = []
for k in ks:
    knn = KNeighborsRegressor(n_neighbors=k)
    knn.fit(X_train, y_train)
    y_pred = knn.predict(X_test)
    test_error = np.sqrt(mean_squared_error(y_test, y_pred))
    test_errors.append(test_error)
    print('K = {}, RMSE on test data = {:.2f}'.format(k, test_error))
```

```
K = 1, RMSE on test data = 0.87
K = 2, RMSE on test data = 0.41
K = 3, RMSE on test data = 1.24
K = 6, RMSE on test data = 1.32
```

Based on these results, it appears that K=2 gives the best performance on the test data, with the lowest test RMSE value of 0.41. Therefore, our choice of K does change based on the new test data.

Q2.

(a) To find the optimal values of β0 and β1, we need to minimize the MSE. The derivative of the MSE with respect to β0 and β1 should be zero for the optimal values. Therefore, we have:

$\partial MSE/\partial \beta0 = (-2/n)\ \Sigma(yi - \beta0 - \beta1xi) = 0$
$\partial MSE/\partial \beta1 = (-2/n)\ \Sigma(xi(yi - \beta0 - \beta1xi)) = 0$

Expanding the first equation, we get:

$\Sigma yi - n\beta 0 - \beta 1\Sigma xi = 0$

Similarly, expanding the second equation, we get:

$\Sigma xiyi - \beta 0\Sigma xi - \beta 1\Sigma(xixi) = 0$

Now, solving these two equations simultaneously, we can obtain the closed-form solution for β0 and β1:

$\beta 1 = [n\Sigma xiyi - \Sigma xi\Sigma yi] / [n\Sigma xi2 - (\Sigma xi)2]$
$\beta 0 = y - \beta 1x$

where y and x are the sample means of the response variable and the predictor variable, respectively.

(b) Using the given data points, we have:

$n = 4$, $\Sigma xi = 10$, $\Sigma yi = 9.5$, $\Sigma xi\text{^}2 = 30$, $\Sigma xiyi = 28$

Substituting these values in the above equations, we get:

$\beta 1 = [4(28) - (10)(9.5)] / [4(30) - (10)\text{^}2] = 0.85$
$\beta 0 = 9.5/4 - (0.85)(10/4) = 0.25$

Thus, the fitted model is $Y = 0.25 + 0.85X$.

To compute R2, we first need to compute the total sum of squares (TSS) and the residual sum of squares (RSS):

mean of Y = 9.5/4 = 2.375

Predicted Y-values: 1.1, 1.95, 2.8, 3.65

$TSS = \Sigma(yi - y)2 = (1-2.375)\text{^}2 + (2-2.375)\text{^}2 + (3-2.375)\text{^}2 + (3.5-2.375)\text{^}2 = 2.525$
$RSS = \Sigma(yi - \hat{y}i)2 = (1-1.1)\text{^}2 + (2-1.95)\text{^}2 + (3-2.8)\text{^}2 + (3..5-3.65)\text{^}2 = 0.075$

where ŷi is the predicted value of yi using the fitted model.

Now, we can compute R2 as:

$R2 = 1 - RSS/TSS = 1 - 0.075/2.525 = 0.97$

The value of R2 is close to 1, indicating that the fitted model explains the variability in the data. The parameter β1 represents the slope of the regression line, which in this case is 0.85. It indicates the change in the response variable Y for every unit change in the predictor variable X.

Q3.

(a) One hot encoding is a technique for representing categorical data as binary vectors. It works by creating a binary vector for each category in a categorical feature, where all elements of the vector are set to 0 except for the one corresponding to the category. One hot encoding can be used in various machine learning tasks such as classification, regression, and clustering.

(b)

(i) One hot encoding is appropriate for the Zipcode feature as it is a categorical feature with discrete values.

(ii) One hot encoding is not appropriate for the Price feature as it is a continuous numerical value.

(iii) One hot encoding is appropriate for the City feature as it is a categorical feature with discrete values.

(iv) One hot encoding is not appropriate for the Name of homeowner feature as each homeowner owns only one home, and it is not a categorical feature with multiple values.

(v) One hot encoding is not appropriate for the Year the house was built feature as it is a numerical feature with a continuous range of values.

Q4.

(a) Example 1: Overfit

(b) Example 2: Good Fit

(c) Example 3: Under Fit

Q5.

(a) False. In linear regression, there are closed-form solutions to obtain the optimal model parameters that minimize the MSE. These solutions can be found using methods such as the normal equations or gradient descent, and do not require trying all possible values for the model parameters.

(b) False. Overfitting occurs when a model performs well on the training data but poorly on new, unseen data, which is often measured by the testing error. Therefore, we can detect overfitting when the testing error is larger than the training error.

(c) False. R2 is used as a measure of how much of the variability in the data is explained by the model and can be greater than 1. R2=1 indicates that the model fits the data perfectly, while R2<1 indicates that the model does not capture all the variability in the data.

(d) True. Multi-linear regression is a special case of polynomial regression where the degree of the polynomial is 1, i.e., the model includes linear terms in the predictors only.

(e) False. As K increases, KNN becomes less likely to overfit the data as it relies on a larger number of nearest neighbors to make predictions, leading to a smoother decision boundary. However, if K is too large, the model may underfit the data and not capture the underlying patterns in the data.