Lab Course Machine Learning
Exercise 7

Prof. Dr. Dr. Lars Schmidt-Thieme,
Mofassir ul Islam Arif
Information Systems and Machine Learning Lab
University of Hildesheim
Submission: 14.12.2019 LearnWeb 3115

December 7, 2019

Instructions

Please following these instructions for solving and submitting the exercise sheet.

1. You should submit two things a) python scripts(zipped) / jupyter notebook and b) a pdf document.
2. In the pdf document you will explain your approach (i.e. how you solved a given problem), and present your results in form of graphs and tables.
3. The submission should be made before the deadline, only through learnweb.
4. Unless explicitly mentioned, you are not allowed to use scikit, sklearn or any other library for solve any part. All implementations must be done yourself.

   This lab has 2 components. 1st is this sheet worth 10 points. 2nd is the lecture that will take place in the lab. You need to be present in the lab in order to get those 5 points

1 Exercise Sheet 7a

Datasets

- 1. Classification Datasets: You can use one of the two datasets (or optionally, both datasets).

  (a) Iris dataset D1: Target attribute class:Iris Setosa, Iris Versicolour, Iris Virginica
  https://archive.ics.uci.edu/ml/datasets/Iris
  (b) Wine Quality called D2: (use winequality-red.csv)
  http://archive.ics.uci.edu/ml/ datasets/Wine+Quality

  Note: Dataset D2 can also be used for a regression problem.

```
predict-knn-reg(q ∈ ℝ^M, 𝒟^train := {(x_1, y_1), ..., (x_N, y_N)} ∈ ℝ^M × ℝ, K ∈ ℕ, d):
  allocate array D of size N
  for n := 1 : N:
    D_n := d(q, x_n)
  C := argmin-k(D, K)
  ŷ := (1/K) Σ_{k=1}^{K} y_{C_k}
  return ŷ
```

Figure 1: KNN for regression

Exercise 1: Implement K-Nearest Neighbor (KNN)

Part A:(10 Points) Your task is to implement KNN algorithm. To implement KNN you have to

- Split data into a train and a test split (70% and 30% respectively).
- Implement a similarity (or a distance) measure. To begin with you can implement the Euclidean Distance
- Implement a function that returns top K Nearest Neighbors for a given query (data point).
- You should provide the prediction for a given query (for a classification task you can use majority voting and for a regression you can use mean).
- Measure the quality of your prediction. [Hint: You have to choose a quality criterion according to the task you are solving i.e. a regression or a classification task **Defend your choice**].

Part B: (5 Points): **Determine Optimal Value of K in KNN algorithm**. In this exercise you have to provide the optimal value of K for given datasets.

- 1. How you can choose value of K for KNN. Give a criterion to choose an optimal value of K
- 2. Implement the criterion for choosing the optimal value of K.
- 3. Experimentally, give evidence that your chosen value is better than other values of K. [Hint: run your experiment with different values of K and plot the error measure for each value].

2 Algorithms

2.1 ANNEX

- You can use numpy
- You can use pandas to read and processing data

```
predict-knn-class(q ∈ ℝ^M, 𝒟^train := {(x_1, y_1), ..., (x_N, y_N)} ∈ ℝ^M × 𝒴, K ∈ ℕ, d):
  allocate array D of size N
  for n := 1 : N:
    D_n := d(q, x_n)
  C := argmin-k(D, K)
  allocate array p̂ of size |𝒴|
  for k := 1 : K:
    p̂_{C_k} := p̂_{C_k} + 1/K
  return p̂
```

Figure 2: KNN for classification

```
argmin-k(x ∈ ℝ^N, K ∈ ℕ) :
  allocate array T of size K
  for n = 1 : min(K, N):
    insert-bottomk(T_{1:n}, n, π_x, 1)
  for n = K + 1 : N:
    if x_n < x_{T_K}:
      insert-bottomk(T, n, π_x, 0)
  return T

insert-bottomk(T ∈ 𝒳^K, n ∈ 𝒳, π : 𝒳 → ℝ, s ∈ ℕ) :
  k := find-sorted(T_{1:K−s}, n, π)
  for l := K : k + 1 decreasing:
    T_l := T_{l−1}
  T_{k+1} := n
```

Note: $\pi_x(n) := x_n$ comparison by x-values. Here, $\mathcal{X} := \mathbb{N}$.

Figure 3: KNN argmin

```
find-sorted-linear(x ∈ 𝒳^K, z ∈ 𝒳, π : X → ℝ) :
  k := K
  while k > 0 and π(z) < π(x_k):
    k := k − 1
  return k
```

- requires
  - x is sorted (increasingly w.r.t. π)
- returns smallest index k with $\pi(x_k) \leq \pi(z)$
  - 0, if $\pi(z) < \pi(x_1)$

Note: Esp. for larger K it is better to use binary search.

Figure 4: KNN sorted

- You can use matplotlib for plotting.
- You should not use any machine learning library for solving the problem i.e. scikit-learn etc. If you use them you will not get any points for the task.