

COMP3308/3608 Artificial Intelligence

Weeks 5 Tutorial exercises

Introduction to Machine Learning. K-Nearest Neighbor and 1R.

No homework this week as planned originally. Although the quiz is cancelled there are many other disruptions and complications for all of us. Some of our exchange and international students are returning home – it is a very stressful time for them. We will re-start the homeworks next week.

Exercise 1. K-Nearest Neighbor with numeric attributes

Consider the dataset given below where *years_experience* is the only attribute and *salary* is the class. What will be the prediction for the new example *years_experience*=5 using 1-Nearest-Neighbor and 3-Nearest-Neighbor with Manhattan distance?

	<i>years_experience</i>	<i>salary</i>
1	4	low
2	9	medium
3	8	medium
4	15	high
5	20	high
6	7	medium
7	12	medium
8	23	medium
9	1	low
10	16	medium

Solution:

The last column of the table shows the Manhattan distance between the new example *years_experience*=5 and each example from the dataset:

	<i>years_experience</i>	<i>salary</i>	D(new,current)
1	4	low	1
2	9	medium	4
3	8	medium	3
4	15	high	10
5	20	high	15
6	7	medium	2
7	12	medium	7
8	23	medium	18
9	1	low	4
10	16	medium	11

The closest neighbor is ex.1 (distance=1) => 1-Nearest-Neighbor will predict *salary*=*low*.

The closest 3 neighbors are ex.1, ex.6 and ex.3 => 3-Nearest-Neighbor will predict *salary*=*medium* (2 *medium* and 1 *low*, the majority prediction is *medium*).

Exercise 2. K-Nearest Neighbor with nominal and numeric attributes

The following dataset describes adults as short, medium and tall based on two attributes: *gender* and *height* in meters.

name	gender	height	class
Cristina	F	1.6	short
Jim	M	2	tall
Margaret	F	1.9	medium
Stephanie	F	1.88	short
Caitlin	F	1.6	short
David	M	1.7	short
William	M	2.2	tall
Stephen	M	2.1	tall
Debbie	F	1.8	medium
Todd	M	1.95	medium

What would be the prediction of 5-Nearest-Neighbor using Euclidian distance for Maria who is (F, 1.75)? Show your calculations. Do not apply normalization for this exercise (but note that in practical situations you need to do this for the numeric attribute!). In case of ties, make random selection.

Hint: *gender* is nominal attribute; see the lecture slides about how to calculate distance for nominal attributes.

Solution:

Gender is a nominal attribute, height is a numeric. When calculating distance for nominal attributes we can use the following rule:

difference = 1 between 2 values that are not the same

difference = 0 between 2 values that are the same

Maria: gender=F, height=1.6

$D(\text{cristina, maria}) = \sqrt{0+(1.6-1.75)^2} = \sqrt{0.0225}$ *short

$D(\text{jim, maria}) = \sqrt{1+(2-1.75)^2} = \sqrt{1.0625}$

$D(\text{margaret, maria}) = \sqrt{0+(1.9-1.75)^2} = \sqrt{0.0225}$ *medium

$D(\text{stephanie, maria}) = \sqrt{0+(1.88-1.75)^2} = \sqrt{0.0169}$ *short

$D(\text{caitlin, maria}) = \sqrt{0+(1.6-1.75)^2} = \sqrt{0.0225}$ *short

$D(\text{david, maria}) = \sqrt{1+(1.7-1.75)^2} = \sqrt{1.0025}$

$D(\text{william, maria}) = \sqrt{1+(2.2-1.75)^2} = \sqrt{1.2025}$

$D(\text{stephen, maria}) = \sqrt{1+(2.1-1.75)^2} = \sqrt{1.1225}$

$D(\text{debbie, maria}) = \sqrt{0+(1.8-1.75)^2} = \sqrt{0.0025}$ *medium

$D(\text{todd, maria}) = \sqrt{1+(1.95-1.75)^2} = \sqrt{1.04}$

The closest 5 neighbors are Debbie, Stephanie, Cristina, Margaret, Caitlin. Of these 5 examples 3 are short and 2 are medium. Thus, 5-nearest neighbor will classify Maria as short.

Note: If there is a tie, the selection is random as the exercises says. For example, if there were 2 *tall*, 2 *medium* and 1 *short*, *tall* or *medium* would be selected randomly.

Exercise 3. 1R algorithm

Consider the *iPhone* dataset given below. There are 4 nominal attributes (age, income, student, and credit_rating) and the class is buys_iPhone with 2 values: yes and no. Predict the class of the following new example using the 1R algorithm: age \leq 30, income=medium, student=yes, credit-rating=fair. Break ties arbitrary.

age	income	student	credit_rating	buys_iPhone
\leq 30	high	no	fair	no
\leq 30	high	no	excellent	no
31..40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31..40	low	yes	excellent	yes
\leq 30	medium	no	fair	no
\leq 30	low	yes	fair	yes
>40	medium	yes	fair	yes
\leq 30	medium	yes	excellent	yes
31..40	medium	no	excellent	yes
31..40	high	yes	fair	yes
>40	medium	no	excellent	no

Dataset adapted from J. Han and M. Kamber, Data Mining, Concepts and Techniques, 2nd edition, 2006, Morgan Kaufmann.

Solution:

1. Attribute age

\leq 30: 3 no, 2 yes; age \leq 30 \rightarrow buys_iPhone=no, errors: 2/5

31..40: 4 yes, 0 no; age = 31..40 \rightarrow buys_iPhone=yes, errors: 0/4

>40: 3 yes, 2 no; age>40 \rightarrow buys_iPhone=yes, errors: 2/5

total errors: 4/14

2. Attribute income

high: 2 no, 2 yes*; income=high \rightarrow buys_iPhone=yes, errors: 2/4

medium: 4 yes, 2 no, income=medium \rightarrow buys_iPhone=yes, errors: 2/6

low: 3 yes, 1 no; income=low \rightarrow buys_iPhone=yes, errors 1/4

total errors: 5/14

3. Attribute student

no: 4 no, 3 yes; student=no \rightarrow buys_iPhone=no, errors: 3/7

yes: 6 yes, 1 no, student=yes \rightarrow buys_iPhone=yes, errors: 1/7

total errors:

4/14

4. Attribute credit_rating

fair: 2 no, 6 yes, credit_rating=fair \rightarrow buys_iPhone=yes, errors 2/8

excellent: 3 no, 3 yes*; credit_rating=yes \rightarrow buys_iPhone=yes, errors 3/6

total errors: 5/14

*- random selection

Rules 1 and 3 have the same (minimum) number of errors; let's choose randomly the first one \Rightarrow

1R produces the following rule:

if age \leq 30 then buys_iPhone=no

```
elseif age=31..40 then buys_iPhone=yes
elseif age>40 then buys_iPhone=yes
```

The new example age≤30, income=medium, student=yes, credit-rating=fair
is classified as buys_iPhone=no.
Note that rule 3 would classify it as buys_iPhone=yes.

Exercise 4. Using Weka

Weka is an open-source ML software written in Java and developed at the University of Waikato in New Zealand, for more information see <http://www.cs.waikato.ac.nz/ml/weka/index.html>. It is a solid and well-established software **written by ML experts**.

Install Weka on your own computer at home, it will be required for Assignment 2. Meanwhile we will use it during the tutorials to better understand the ML algorithms we study.

1. Get familiar with Weka (Explorer). Load the weather data with nominal attributes (weather.nominal.arff). The datasets are in \Program Files\Weka-3.8.
2. Choose ZeroR classifier.
3. Choose “Percentage split” as “Test option” for evaluation of the classifier: 66% training set, 33% testing set

ZeroR is a very simple ML algorithm (also called “straw man”). It works as follows:

- Nominal class (classification) – determine the most frequent class in the training data *most_freq*; classification: for every new example (i.e. not in the training data), return *most_freq*
- Numeric class (regression) – determine the average class value in the training data *av_value*; classification: for every new example (i.e. not in the training data), return *av_value*

ZeroR is useful to determine the baseline accuracy when comparing with other ML algorithms.

4. Run the IBk algorithm (k-nearest neighbor, IB comes from Instance-Based learning; it is under “Lazy”) with k=1 and 3.

Was IB1 more accurate than IB3 on the test data? Does a higher k mean better accuracy?
How does IB1 and IB3 compare with ZeroR in terms of accuracy on test set?

5. Consider IB1. Change “Percentage split” to “Use training data”(i.e. you will build the classifier using the training data and then test it on the same set). What is the accuracy? Why? Would you achieve the same accuracy if IB1 was IBk, for k different than 1?

6. Now load another dataset – iris.arff. Run and compare the same algorithms. Don’t forget to normalize the data (the attributes are numeric). The normalization feature is under Filters-Unsupervised-Attribute-Normalise.

Why we didn’t normalize the weather data when we run IBLk above on the weather data?

Answer: The attributes were nominal; Weka calculates the distance as in exercise 2 (0 if the attribute values are the same, 1 if they are different) => all attributes are on the same scale, no need to normalize.

7. Select the K-star algorithm, another lazy algorithm, a modification of the standard nearest neighbor. Read its short description (right click on the algorithm’s name -> “Show properties” -> “More”). Run it and compare the results (accuracy on test set using “Percentage split” again) with the 1 and 3 nearest neighbor algorithms and ZeroR.

8. Now run the 1R algorithm on both the iris and weather.nominal datasets using the same evaluation procedure (“Percentage split”). 1R is under “rules” in Weka and called “OneR”.

What are the rules that 1R formed? How many rules? How many attributes in each rule? Which are these attributes for the two datasets?

Answer: 1R produces 1 rule, which tests the values of 1 attribute. For example, for the iris dataset the 1R rule is based on *petalwidth*, and for the weather dataset – on *outlook*:

petalwidth:

< 0.8 -> Iris-setosa
 < 1.75 -> Iris-versicolor
 >= 1.75 -> Iris-virginica

outlook:

sunny -> no
 overcast -> yes
 rainy -> yes

9. Compare the accuracy of ZeroR, OneR, 1R, IB1, IBk on the test set. Which was the most accurate classifier?