UNIVERSITY OF SASKATCHEWAN

Department of Computer Science
176 Thorvaldson Building
110 Science Place, Saskatoon, SK, S7N 5C9, Canada
Telephone: (306) 966-4886, Facimile: (306) 966-4884

CMPT 423/820
Winter 2021
Machine Learning

Assignment 2

Simple Classifiers

Date Due: October 19, 2021, 5pm **Total Marks: 60**

Version History

- **5/10/2021**: released to students

General Instructions

- **This assignment is individual work.** You may discuss questions and problems with anyone, but the work you hand in for this assignment must be your own work.
- Each question indicates what to hand in.
- **Assignments must be submitted to Canvas.**
- Assignments will be accepted until 11:59pm without penalty.

Software

This assignment primarily exercises the use of Scikit-Learn, an Open Source collection of tools for Machine Learning. This collection is quite extensive, and could be intimidating. Rest assured that we will practice our understanding of concepts with this tool, but you won't be tested on how well you know the software.

You'll use Jupyter Notebooks to complete these assignment questions, and you will be allowed to make use of all the software we introduced in A1. As you complete this assignment, you may wonder if you can use this or that package that you might have heard about. The answer will depend on how that package contributes to your work. If the software you are considering accomplishes the learning objectives in a way that means you don't have to think about them, I will probably disallow it. If the software does not impact your attention to the learning objectives, I will probably allow it. It's always good to ask.

As in A1, you'll be asked to submit a PDF version along with your Jupyter Notebooks. This will assist the markers. Marks will be deducted if PDFs are not submitted. I have found that the most reliable way to produce a PDF of a Jupyter Notebook is to Export to HTML, Open the HTML in a browser, and Print As PDF from your browser.

Question 1 (15 points):

Learning Objectives:

- Practical experience building Naive Bayes Classifiers
- Practice critically evaluating the performance of classifiers.

Competency Level: Basic

The IRIS dataset (used in A1 and also in lecture) has 4 continuous features/attributes, and the class label. We saw in class that we can get pretty good accuracy using one Gaussian Naive Bayes Classifier (GNBC) when all 4 features/attributes are used.

In this question, we will build 4 different GNBCs, each classifier using only one of the features/attributes to fit the model. In other words, the first classifier will use feature/attribute/column 1, the second classifier will use feature/attribute/column 2, etc.

1. Build the four 1-feature classifiers, and calculate the *accuracy* of each.
2. Build the 4-feature classifier (as we saw in class), and calculate the accuracy.
3. Reproduce the density plots from A1Q6 Task 5 that shows the class density for each feature, and compare the density plots to the accuracy scores you obtained. In a few sentences discuss how the density plot relates to the accuracy score.
4. Compare the best 1-feature classifier to the 4-feature classifier, in terms of accuracy. Discuss briefly your results.

Errata

1. None so far!

What to Hand In

A PDF document exported from Jupyter Notebook, containing the tasks and discussions above, with your name and student number at the top of the document, as in Assignment 1.

- Make sure that your document is well-structured, using headings and providing discussion in Markdown cells.
- Make sure that the markers can read your document and grade it easily.

Evaluation

- You constructed the four 1-feature classifiers, and calculated their accuracies.
  - 3 marks. Your Python scripting was neat and presentable. You made good use of Python comments, and Markdown cells to explain your method to a reader.
  - 2 marks. You calculated the accuracies correctly, and presented them neatly.
- You discussed the relation between accuracy of each 1-feature classifier, and the graphical visualization provided by the class density for each feature.
  - 4 marks. Your discussion highlighted the visual clues that might indicate differences in accuracy.
  - 2 marks. Your discussion was not too long! Seriously, keep it to the point.
- You compared the best 1-feature classifier to the 4-feature classifier in terms of accuracy.
  - 2 marks. Your discussion was relevant.
  - 2 marks. Your discussion was not too long.

Question 2 (10 points):

Learning Objectives:
- Critically assess a dataset based on visualization.

Competency Level: Basic

This question is preparation for Question 3. It's a separate question to prevent your answer for Q3 from being too cluttered.

On the Assignment Moodle page, you'll find a dataset named `A2Q2.cvs`. This dataset has 14 columns. The *first* column is the class label, using the integers 1-3 as labels. The remaining columns are continuous features.

Plot the class densities for all 13 features, similar to A1Q6 Task 5. Comment on each feature, relating the visualization to its potential utility in a classifier (based on your experience from Q1).

Answer the following questions:

- Which, if any, of the 13 features, would you pick as the single feature in a 1-feature classifier? Briefly explain your answer.
- Prior to building a classifier, do you think a classifier based on this data will have high accuracy? Briefly explain your answer.

Errata

1. None so far!

What to Hand In

A PDF document exported from Jupyter Notebook, containing 13 density plots, and brief discussion, with your name and student number at the top of the document, as in Assignment 1.

- Make sure that your document is well-structured, using headings and providing discussion in Markdown cells.
- Make sure the answers to the questions are easy to find!
- Make sure that the markers can read your document and grade it easily.

Evaluation

- 4 marks: Your density plots are correct, and neatly presented.
- 6 marks: You answers to the questions demonstrate you've assessed the features critically.

Question 3 (15 points):

Learning Objectives:
- To critically compare different models on the same dataset.

Competency Level: Basic

On the Assignment Moodle page, you'll find a dataset named `a2q3.cvs`. This dataset has 14 columns. The *first* column is the class label, using the integers 1-3 as labels. The remaining columns are continuous features. Use this dataset to compare three classifiers:

1. K-Nearest Neighbours Classifier. Remember that you'll have to choose K.
2. Naive Bayes Classifier
3. Decision Tree Classifier.

To keep things interesting, use f_1 as the metric for comparison.

Discuss the performance of the three classifiers. Which, if any, would you choose as the best model for the data? Explain your answer.

To complete this question, you'll have to research the Scikit-Learn User Manual to use KNN and Decision Trees.

Errata

1. None so far!

What to Hand In

A PDF document exported from Jupyter Notebook, with your name and student number at the top of the document, as in Assignment 1.

- Make sure that your document is well-structured, using headings.
- Make sure that you've used the Scikit-Learn models correctly.
- Document any decisions about parameter choices, etc, in Markdown cells close to your scripts.
- Address the discussion comparing classifiers in Markdown cells at the end of your document.
- Make sure the different parts of your solution to the question are easy to find!
- Make sure that the markers can read your document and grade it easily.

Evaluation

- 3 marks: You fitted the KNN classifier appropriately by choosing k, and other parameters to the model.
- 3 marks: You fitted the Decision Tree classifier appropriately by choosing appropriate parameters to the model.
- 1 mark: You fitted the Naive Bayes classifier appropriately.
- 8 marks: Your discussion of the performance of the classifiers arrived at a well-reasoned conclusion.

Question 4 (5 points):

Purpose: To exercise the derivation of formulae involving probability.

Degree of Difficulty: Moderate. The actual derivation is easier than the explanation.

References: Lecture Notes 04, Math with LaTeX in Jupyter Notebook

There is a notion in Bayesian statistics that yesterday's posterior probabilities are today's prior probabilities. It's an idea that suggests that learning should naturally combine data collected over time. It also reassures us that choosing a prior can be based on data seen previously. To understand this notion we need to do some math.

Task Suppose we collected data $\mathbf{X}_1$ yesterday, and used the data to calculate $P(y|\mathbf{X}_1)$. Yesterday, the prior that we assumed was $P(\mu) = \text{Beta}(\mu|a,b)$. Today, we collected data $\mathbf{X}_2$, and we wish to calculate $P(y|\mathbf{X}_1, \mathbf{X}_2)$.

Derive an expression for $P(\mu|\mathbf{X}_1, \mathbf{X}_2)$ in terms of yesterday's posterior $P(\mu|\mathbf{X}_1)$. This expression shows how yesterday's posterior can be used as if it were a prior.

Elaboration *In the following, we'll start with a review of the lecture material. Then we'll think about what happens with data $\mathbf{X}_1$ collected yesterday, and then more data $\mathbf{X}_2$ today. We could just throw away the model based on $\mathbf{X}_1$, and start over with all the data. But we can be cleverer than that here.*

In class we derived the following equation using Bayes' Rule:

$$P(\mu|\mathbf{X}) = \frac{P(\mathbf{X}|\mu)P(\mu)}{P(\mathbf{X})}$$

This was one of the steps in determining $P(y|\mathbf{X})$ for a binary event Y. In this expression, $P(\mu)$ is the prior distribution for μ, and $P(\mu|\mathbf{X})$ is the posterior. We assumed that $P(\mu) = \text{Beta}(\mu|a,b)$, and we learned that $E[\mu] = \frac{a}{a+b}$. We also saw (skipping some of the mathematical details) that:

$$E[\mu|\mathbf{X}] = \frac{m+a}{N+a+b}$$

where m and N come from the data $\mathbf{X}$. The posterior $P(\mu|\mathbf{X})$ turns out also to be a Beta distribution over μ, but it's $\text{Beta}(a_1, b_1)$, where $a_1 = m + a$ and $b_1 = N - m + b$ are new hyper-parameters. In effect, $\text{Beta}(\text{a}_1, \text{b}_1)$ summarizes everything we learned about μ from $\mathbf{X}$.

Suppose we collected data $\mathbf{X}_1$ yesterday, and used the data to calculate $P(y|\mathbf{X}_1)$. Yesterday, the prior that we assumed was $P(\mu) = \text{Beta}(\mu|a,b)$. Today, we collected data $\mathbf{X}_2$, and we wish to calculate $P(y|\mathbf{X}_1, \mathbf{X}_2)$.

In class, we were able to show, by significant hand-waving, that

$$P(y|\mathbf{X}_1, \mathbf{X}_2) = \frac{m_1 + m_2 + a}{N_1 + N_2 + a + b}$$

This is exactly what we'd get if we combined $\mathbf{X}_1, \mathbf{X}_2$ together. But it also shows that we can update our uncertainty about y, by counting m_2, N_2 in $\mathbf{X}_2$, and re-using the counts m_1, N_1 from yesterday. Your work in this question replaces some of the handwaving with a more formal derivation.

Hint: There's no calculus involved, no tricky business with Beta, Bernoulli, or Binomial, only the basic rules of probability: product rule, Bayes rule, conditional independence, etc. Start with Bayes Rule. This is a lot easier than it seems.

One possibly confusing idea is that $P(\mu|\mathbf{X}_1)$ doesn't look like a prior, because it is conditional. That's true. But in this context, we can look at the prior more in terms of its use, than by its notational syntax.

Errata

1. None so far!

What to Hand In

- A Jupyter Notebook named `A2Q4.ipynb` with your derivation encoded in markdown, using LaTeX for the math.
- A PDF document `A2Q4.pdf` exported from Jupyter Notebook, containing the derivation above. The marker will primarily look at this, so make it presentable, like a report.

Be sure to include your name, NSID, student number, and course number at the top of all documents.

Evaluation

- 5 marks: Your derivation has $P(\mu|\mathbf{X}_1, \mathbf{X}_2)$ on the left hand side, $P(\mu|X)$ on the right hand side, and your derivation applied valid rules of probability.

[redacted]

Question 5 (15 points):

Learning Objectives: • To use theoretical principles to adapt software implementations of Naive Bayes to handle mixed data.

Competency Level: Advanced

Currently, Scikit-Learn has 4 implementations of Naive Bayes. Each implementation assumes that all the features have the same kind of feature distribution. For example, the Scikit-Learn implementation of Gaussian Naive Bayes Classifier assumes that all features are numeric, and the histogram of the features given the class label are more-or-less bell shaped around a mean value. On the other hand, the Scikit-Learn implementation of the Categorical Naive Bayes Classifier assumes that all features are categorical. **This is a limitation of the software, not a theoretical limitation of Naive Bayes.**

In this question, you are invited to explain, or describe how we could use these two classifiers to handle mixed data.

This is an open-ended question. You could address it at a number of different levels of detail.

1. Informally. You can describe what you would do without going into a lot math or Python scripting. This could show that you have some ideas, and the ideas well-considered.
2. Formally. You can start with the Naive Bayes Formula, and produce a revised version that shows how two different Naive Bayes classifiers can be combined. This would show that you were able to take your informal ideas and derive a formula that correctly shows how it can be done.
3. Practically: You can take your informal or formal basis, and give a demonstration of the basis in terms of the software provided by Scikit-Learn. This would show that you have substantiated your ideas empirically.

You can decide how you want to answer this question. If you have time, or interest, you can do more work, for more marks. If you have less time, then address it informally, for less marks.

Note: An informal description is valuable, even though it is worth less marks. Your choice here is a compromise between time, and effort. It's not a failure to give an informal answer.

Errata

1. None so far!

What to Hand In

A PDF document exported from Jupyter Notebook, with your name and student number at the top of the document, as in Assignment 1.

- Make sure that your document is well-structured, using headings, LaTeX math as appropriate to your answer, and Python demonstrations, if you get to that.

Evaluation

For this question, we will apply the following rubric.

- Zero marks: You submitted nothing.
- 10 marks: Your approach.
  - You submitted an informal description of an approach.
    - 6/10 marks: Your description is clear, and your approach will lead to a correct combination.
    - 3/10 marks: Your description was vague, or does not lead to a correct combination.

  - You submitted a formal derivation.
    - 10/10 marks: Your formal derivation is completely correct.
    - 7/10 marks: Your derivation didn't get to the right answer, or made mathematical errors along the way.

- 5 marks: You were able to demonstrate your approach in software, using a mixed dataset.