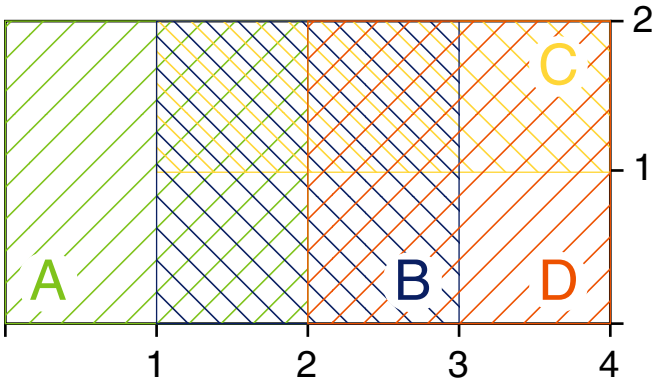
Lab Class ML:IV

By Wednesday, 2017-12-20 solutions for the following exercises have to be submitted: 1, 2, 5, and 6.

Exercise 1 : Probability Basics (Conditional Independence)

You're throwing darts at a 2x4m wall with the overlapping rectangular regions marked A (2x2m), B (2x2m), C (1x3m) and D (2x2m) in the illustration. Assume that you're guaranteed to hit the wall on every throw, and that the probability of hitting any particular point on the wall is the same for all points.



Let A refer to the event of the dart hitting an arbitrary point inside the region marked A; and B, C, D to the respective events for the other regions. Determine the following probabilities:

(a) $P(A)$, $P(B)$, $P(C)$, and $P(D)$.

(b) $P(A \cap B)$, $P(A \cap C)$, $P(B \cap C)$, and $P(B \cap D)$.

(c) Check all that apply:

- ☐ The events A and B are statistically independent.
- ☐ The events A and C are statistically independent.
- ☐ The events B and C are statistically independent.
- ☐ The events B and D are statistically independent.

(d) $P(A \mid C)$, $P(B \mid C)$, and $P(A \cap B \mid C)$.

(e) $P(B \mid D)$, $P(C \mid D)$, and $P(B \cap C \mid D)$

(f) Check all that apply:

- ☐ The events A and B are conditionally independent given C.
- ☐ The events B and C are conditionally independent gived D.

Exercise 2 : Probability Basics (Kolmogorov)

Let B be an event with probability $P(B) > 0$, and f a real-valued function with $\mathsf{f}(X) = P(X \mid B)$. Show that f is a probability measure.

Exercise 3 : Probability Basics

Which of the following statements are true?

- ☐ According to the Kolmogorov axioms the statement $P(A) - P(\overline{A}) = 0$ holds.
- ☐ A function that fulfills the Kolmogorov axioms is a probability measure.
- ☐ Two events are statistically independent $\Leftrightarrow P(A \cap B) = P(A) + P(B)$.
- ☐ Each subset A of a sample space Ω is an event.

Exercise 4 : Bayes

The formula $\underset{B_i \in \{B_1,\ldots,B_k\}}{\text{argmax}} P(B_i) \cdot \prod_{j=1}^{p} P(A_j \mid B_i)$ evaluates to

- ☐ a probability
- ☐ the most likely B_i, if the naive Bayes assumption holds
- ☐ a ranking
- ☐ a positive number

Exercise 5 : Bayes

A hospital database contains diagnoses (diseases) along with observed symptoms, collected during the past years. Let following representative dump be given, where the diseases are sorted temporally according to their appearances. Note, that the symptoms for a disease can change over different time periods.

		Symptom	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9
Year	Diagnosis										
2001	D_1		X		X		X				
2002	D_2			X		X	X		X		
2003	D_3		X		X			X		X	
2004	D_4			X		X	X		X		
2005	D_3		X		X					X	
2006	D_5						X				X
2007	D_3		X		X			X			
2008	D_2			X					X		

(a) Compute the prior probabilities $P(D_i)$.

(b) Compute the posterior probabilities $P(D_i \mid S_4)$ of the diagnoses D_i given symptom S_4.

Exercise 6 : P Content-based Spam Filtering with Naive Bayes

Your task is to implement a simple content-based spam filter using the Naive Bayes approach: given a set of spam and non-spam ("ham") emails, your program should learn to compute the probability that a previously unseen email is spam based on the occurrence of words and other tokens. For the purpose of our implementation, we will consider the presence or absence of words w_i in an email as binary events, which are assumed to be statistically independent according to the Naive Bayes assumption discussed in the lecture.

For training and testing our classifier, we will use the SpamAssassin public mail corpus.

(a) Read the article "A Plan for Spam" from http://paulgraham.com/antispam.html.

(b) Download the following two datasets of spam and non-spam emails:

- http://spamassassin.apache.org/old/publiccorpus/20050311_spam_2.tar.bz2 (containing $\approx$ 1400 spam emails)
- http://spamassassin.apache.org/old/publiccorpus/20030228_easy_ham_2.tar.bz2 (containing $\approx$ 1400 ham emails)

The example emails are provided as individual files within a gzipped tar archive, many of them using the Quoted-printable MIME encoding. With the help of the Python standard library, write a set of utility functions for extracting and decoding the individual emails, and tokenizing them into their constituent words. (Hints will be given in lab class)

(c) Develop a class `SpamClassifier` which implements a simple Naive Bayes spam filter. The constructor of your class should take two parameters: a collection of spam emails, and a collection of ham emails—both already split into their constituent tokens—which are then used to train the classifier. During training, your classifier should estimate the probability $P(w_i|\mathsf{spam})$, for each word w_i which occurs in either a spam or a non-spam email. Also consider how you deal with words that do not occur in the training data. Your class should provide a method `predict`, which takes the words w_j of a single email as an argument, and returns the probability $P(\mathsf{spam}|w_j)$ that this email is spam.

The interface to your classifier should work as follows:

```
>>> cls = SpamClassifier([["this", "is", "spam"], ["more", "spam"]],
                         [["this", "is", "ham"], ["more", "ham"]])
>>> cls.predict(["is", "this", "spam", "or", "not"])
0.92
```

(d) Train and test your spam filter using the SpamAssassin corpus. Adapt your previous implementation of cross-validation to work with your spam classifier. Select a threshold for the predicted spam probability, above which you classify an email as spam. Using 10 cross-validation folds, compute the average misclassification rate, separately for the `spam` and `ham` classes. Which probability threshold works best for this dataset?

(e) Train your classifier on the entire dataset (without cross-validation) and examine the conditional spam probabilities for individual words that your classifier computes during training. Which words are the strongest evidence that an email is spam? Which words provide the weakest evidence for either class?

Test the trained classifier using the following additional datasets:

- https://spamassassin.apache.org/old/publiccorpus/20030228_spam.tar.bz2 and
- https://spamassassin.apache.org/old/publiccorpus/20030228_hard_ham.tar.bz2

What is the misclassification performance now?

(f) Discuss the Naive Bayes assumption in the context of text classification. Does the assumption hold?