CSCI 544 – Applied Natural Language Processing, Spring 2020

Written Homework 1: Naive Bayes, Linear Classifiers, Perceptron

Out: January 22, 2020

Total: 18 pages.

General instructions

1. This is not a graded assignment. Do not turn it in.
2. The assignment is meant as preparation for the in-class exams. You should aim to answer all the questions on your own, without help.
3. Space is provided as it would be on the exams. Answers should be concise and fit in the space provided; in the exams it will not be possible to add space, and long and rambling answers will be penalized.
4. After solving the problems (or giving them your best try), you are encouraged to discuss and compare solutions with your classmates.
5. You are welcome to discuss the problems with us. We encourage open discussion on Piazza, so that the entire class can benefit from the discussion.
6. ~~Answers to select problems will be distributed at a later time.~~
   Answers are indicated in blue.

Problem 1. You are building a classifier for the sentiment of Russian adjectives. The following 100 adjectives have been sampled from the class of positive adjectives, to use as training data. The adjectives have been analyzed into a stem and suffix.

Adjective	Stem + suffix	Count
красивый	красив + ый	10
красивая	красив + ая	18
красивую	красив + ую	12
приятный	приятн + ый	10
приятная	приятн + ая	32
приятную	приятн + ую	18

a. Based on the training data, give estimates for the probabilities of the individual stems and suffixes below.

P(красив- | positive) = *0.4 ($\frac{40}{100}$)* P(приятн- | positive) = *0.6 ($\frac{60}{100}$)*

P(-ый | positive) = *0.2 ($\frac{20}{100}$)* P(-ая | positive) = *0.5 ($\frac{50}{100}$)* P(-ую | positive) = *0.3 ($\frac{30}{100}$)*

b. Suppose that the stem and suffix are conditionally independent, given the class (that is, a naive Bayes model). If the probability estimates you just calculated exactly describe the class of positive adjectives, how many instances of each word would you expect to find in a sample of 100 words drawn from the class of positive adjectives?

красивый	*8*	*(0.4×0.2×100)*
красивая	*20*	*(0.4×0.5×100)*
красивую	*12*	*(0.4×0.3×100)*
приятный	*12*	*(0.6×0.2×100)*
приятная	*30*	*(0.6×0.5×100)*
приятную	*18*	*(0.6×0.3×100)*

c. Is it possible to construct any sort of model that better fits the observed sample? If so, how? If not, why not?

Yes, it is possible to model the joint probability of stem+suffix directly from the observed sample, without assuming conditional independence.

d. Roughly speaking (without calculating numbers), does our observed sample provide strong evidence against using a naive Bayes model for describing the class of positive adjectives? Why or why not?

No, the sample does not provide strong evidence against using a naive Bayes model, because the fit of the model to the observed data is fairly good, and the discrepancies can be a result of random sampling.

Problem 2. In this problem you will use probabilities to segment Arabic words into prefixes, stems and suffixes. Since we are able to give little data about stems, we concentrate only on prefixes and suffixes. The following segmented words are used as training data (∅ denotes a null prefix or suffix).

Arabic script	Prefix + Stem + Suffix	Meaning
لولده	l + wld + h	to his child
وعدك	∅ + wʕd + k	your promise
وكتبه	w + ktb + h	and his books
فكتبي	f + ktb + y	and my books
فعمله	f + ʕml + h	and his work
لعملك	l + ʕml + k	to your work
وشغل	w + šɣl + ∅	and work
باذنه	b + Aðn + h	with his permission
صحتك	∅ + SHt + k	your health
فابني	f + Abn + y	and my son

a. Give the maximum likelihood estimates for the probability of each prefix (don't forget the null prefix):

l: 0.2 w: 0.2 f: 0.3 ∅: 0.2 b: 0.1

b. Give the maximum likelihood estimates for the probability of each suffix (don't forget the null suffix):

h: 0.4 ∅: 0.1 k: 0.3 y: 0.2

c. For segmenting words, we make the simplifying assumption that any sequence of characters is possible and equally likely as a stem; however, we do impose a constraint that a stem is at minimum three characters. Given this constraint, find the most likely segmentation for each word (use the transliteration, not the Arabic characters):

Arabic	Transliteration	Prefix + Stem + Suffix	Likelihood of prefix+stem+suffix
فعلي	fʕly	*∅ + fʕl + y*	*0.2 × 0.2 = 0.04*
وضحك	wDHk	*∅ + wDH + k*	*0.2 × 0.3 = 0.06*

d. Does the segmenter always give the most common prefix that is consistent with the beginning of the word? Why or why not?

The segmenter does not always give the most common prefix: for the word f'ly it chose the prefix ∅, even though f is more common. The reason is that choosing the most common prefix f and most common suffix y would violate the constraint on minimum stem length; to satisfy the constraint, either the prefix or the suffix (or both) must be ∅.

e. Does the segmenter always give the most common suffix that is consistent with the ending of the word? Why or why not?

For 4-character words the segmenter always chooses the most common suffix, because the alternative ∅ has such low probability that it's always better to give up a potential prefix than a suffix. However, for 3-letter words the constraint on stem length would force the stemmer to choose the suffix ∅.

Problem 3. Named entity recognition (NER) is the problem of identifying the names of persons, organizations, locations etc. In this problem you will construct a naive Bayes classifier to identify named entities in Czech. The table below is a snapshot of the data set, where phrases are labeled as to whether or not they represent a named entity. Each phrase is followed by the number of times it appears in the data.

Named entities	Not named entities
Nové Město (3)	Nové Auto (1)
Nové Dillí (5)	Kostel (9)
Kostel Panny Marie (2)	Červený (7)
Pan Červený (1)	Staré Auto (3)
Marie (4)	Nové (12)
	Červený Muž (3)

a. Identify the priors for each class:

Named entity: $\frac{15}{50} = 0.3$ Not named entity: $\frac{35}{50} = 0.7$

b. You will be constructing two types of features: *first word*, and *any word*. The *first word* feature of a phrase is the first word of the phrase; the *any word* feature of a phrase will have multiple occurrences – one for each word, including the first (so a three-word phrase, for example, will have three *any word* features).

Start by tabulating the number of instances of each feature, for each class.

	First word		Any word	
	Named Entity	Not Named Entity	Named Entity	Not Named Entity
Červený	*0*	*10*	*1*	*10*
Kostel	*2*	*9*	*2*	*9*
Marie	*4*	*0*	*6*	*0*
Nové	*8*	*13*	*8*	*13*
Pan	*1*	*0*	*1*	*0*
Staré	*0*	*3*	*0*	*3*
Auto			*0*	*4*
Dillí			*5*	*0*
Město			*3*	*0*
Muž			*0*	*3*
Panny			*2*	*0*

c. Apply Laplace (add-one) smoothing, and calculate the probabilities of each feature, conditional upon class.

	First word		Any word	
	Named Entity	Not Named Entity	Named Entity	Not Named Entity
Červený	$\frac{1}{21} = 0.048$	$\frac{11}{41} = 0.268$	$\frac{2}{39} = 0.051$	$\frac{11}{53} = 0.208$
Kostel	$\frac{3}{21} = 0.143$	$\frac{10}{41} = 0.244$	$\frac{3}{39} = 0.077$	$\frac{10}{53} = 0.189$
Marie	$\frac{5}{21} = 0.238$	$\frac{1}{41} = 0.024$	$\frac{7}{39} = 0.179$	$\frac{1}{53} = 0.019$
Nové	$\frac{9}{21} = 0.429$	$\frac{14}{41} = 0.341$	$\frac{9}{39} = 0.231$	$\frac{14}{53} = 0.264$
Pan	$\frac{2}{21} = 0.095$	$\frac{1}{41} = 0.024$	$\frac{2}{39} = 0.051$	$\frac{1}{53} = 0.019$
Staré	$\frac{1}{21} = 0.048$	$\frac{4}{41} = 0.098$	$\frac{1}{39} = 0.026$	$\frac{4}{53} = 0.075$
Auto			$\frac{1}{39} = 0.026$	$\frac{5}{53} = 0.094$
Dillí			$\frac{6}{39} = 0.154$	$\frac{1}{53} = 0.019$
Město			$\frac{4}{39} = 0.103$	$\frac{1}{53} = 0.019$
Muž			$\frac{1}{39} = 0.026$	$\frac{4}{53} = 0.075$
Panny			$\frac{3}{39} = 0.077$	$\frac{1}{53} = 0.019$

Applying Laplace smoothing in the blank cells is acceptable, in which case the values in the first two columns should be as follows:

Červený	$\frac{1}{26} = 0.038$	$\frac{11}{46} = 0.239$
Kostel	$\frac{3}{26} = 0.115$	$\frac{10}{46} = 0.217$
Marie	$\frac{5}{26} = 0.192$	$\frac{1}{46} = 0.022$
Nové	$\frac{9}{26} = 0.346$	$\frac{14}{46} = 0.304$
Pan	$\frac{2}{26} = 0.077$	$\frac{1}{46} = 0.022$
Staré	$\frac{1}{26} = 0.038$	$\frac{4}{46} = 0.087$
Auto	$\frac{1}{26} = 0.038$	$\frac{1}{46} = 0.022$
Dillí	$\frac{1}{26} = 0.038$	$\frac{1}{46} = 0.022$
Město	$\frac{1}{26} = 0.038$	$\frac{1}{46} = 0.022$
Muž	$\frac{1}{26} = 0.038$	$\frac{1}{46} = 0.022$
Panny	$\frac{1}{26} = 0.038$	$\frac{1}{46} = 0.022$

d. Use your classifier to predict for each of the following phrases whether or not they are a named entity: for each phrase, calculate the probability that it belongs to each class, and then select the most probable class. (Some of the phrases below are not proper Czech; don't worry about it for this exercise.)

	P(Named Entity)	P(Not Named Entity)	Chosen label
Červený Kostel	$0.3 \cdot \frac{1}{21} \cdot \frac{2}{39} \cdot \frac{3}{39} = 5.635 \times 10^{-5}$	$0.7 \cdot \frac{11}{41} \cdot \frac{11}{53} \cdot \frac{10}{53} = 7.354 \times 10^{-3}$	*not NE*
Červený Město	$0.3 \cdot \frac{1}{21} \cdot \frac{2}{39} \cdot \frac{4}{39} = 7.514 \times 10^{-5}$	$0.7 \cdot \frac{11}{41} \cdot \frac{11}{53} \cdot \frac{1}{53} = 7.354 \times 10^{-4}$	*not NE*
Dillí	$0.3 \cdot \frac{6}{39} = 4.615 \times 10^{-2}$	$0.7 \cdot \frac{1}{53} = 1.321 \times 10^{-2}$	*NE*
Kostel Panny Dillí	$0.3 \cdot \frac{3}{21} \cdot \frac{3}{39} \cdot \frac{3}{39} \cdot \frac{6}{39} = 3.901 \times 10^{-5}$	$0.7 \cdot \frac{10}{41} \cdot \frac{10}{53} \cdot \frac{1}{53} \cdot \frac{1}{53} = 1.147 \times 10^{-5}$	*NE*
Pan Auto	$0.3 \cdot \frac{2}{21} \cdot \frac{2}{39} \cdot \frac{1}{39} = 3.757 \times 10^{-5}$	$0.7 \cdot \frac{1}{41} \cdot \frac{1}{53} \cdot \frac{5}{53} = 3.039 \times 10^{-5}$	*NE*
Panny Marie	$0.3 \cdot \frac{3}{39} \cdot \frac{7}{39} = 4.142 \times 10^{-3}$	$0.7 \cdot \frac{1}{53} \cdot \frac{1}{53} = 2.492 \times 10^{-4}$	*NE*
Nové Kostel	$0.3 \cdot \frac{9}{21} \cdot \frac{9}{39} \cdot \frac{3}{39} = 2.282 \times 10^{-3}$	$0.7 \cdot \frac{14}{41} \cdot \frac{14}{53} \cdot \frac{10}{53} = 1.191 \times 10^{-2}$	*not NE*
Nové Marie	$0.3 \cdot \frac{9}{21} \cdot \frac{9}{39} \cdot \frac{7}{39} = 5.325 \times 10^{-3}$	$0.7 \cdot \frac{14}{41} \cdot \frac{14}{53} \cdot \frac{1}{53} = 1.191 \times 10^{-3}$	*NE*
Nové Město	$0.3 \cdot \frac{9}{21} \cdot \frac{9}{39} \cdot \frac{4}{39} = 3.043 \times 10^{-3}$	$0.7 \cdot \frac{14}{41} \cdot \frac{14}{53} \cdot \frac{1}{53} = 1.191 \times 10^{-3}$	*NE*
Staré Dillí	$0.3 \cdot \frac{1}{21} \cdot \frac{1}{39} \cdot \frac{6}{39} = 5.635 \times 10^{-5}$	$0.7 \cdot \frac{4}{41} \cdot \frac{4}{53} \cdot \frac{1}{53} = 9.725 \times 10^{-5}$	*not NE*

It is OK to scale the probabilities so that each row sums up to 1; in this case the numbers should be as in the right-hand side of the table below.
If the person applied Laplace smoothing on all the words in the first two columns in part (c) then the probabilities should be as in the left-hand side of the table below, or as in the middle columns if scaled to 1.

P(NE)	*P(not NE)*	*P(NE)*	*P(not NE)*		*P(NE)*	*P(not NE)*
4.552×10^{-5}	6.555×10^{-3}	*0.00690*	*0.99310*	*Červený Kostel*	*0.00760*	*0.99240*
6.069×10^{-5}	6.555×10^{-4}	*0.08474*	*0.91526*	*Červený Město*	*0.09270*	*0.90730*
1.775×10^{-3}	2.871×10^{-4}	*0.86077*	*0.13923*	*Dillí*	*0.77751*	*0.22249*
3.151×10^{-5}	1.022×10^{-5}	*0.75507*	*0.24493*	*Kostel Panny Dillí*	*0.77283*	*0.22717*
3.034×10^{-5}	2.709×10^{-5}	*0.52836*	*0.47164*	*Pan Auto*	*0.55282*	*0.44718*
1.593×10^{-4}	5.417×10^{-6}	*0.96711*	*0.03289*	*Panny Marie*	*0.94325*	*0.05675*
1.843×10^{-3}	1.062×10^{-2}	*0.14793*	*0.85207*	*Nové Kostel*	*0.16078*	*0.83922*
4.301×10^{-3}	1.062×10^{-3}	*0.80202*	*0.19798*	*Nové Marie*	*0.81719*	*0.18281*
2.458×10^{-3}	1.062×10^{-3}	*0.69833*	*0.30167*	*Nové Město*	*0.71866*	*0.28134*
4.552×10^{-5}	8.668×10^{-5}	*0.34432*	*0.65568*	*Staré Dillí*	*0.36688*	*0.63312*

e. Why do we construct the feature as "any word" rather than "word other than first"? (Hint: how would we classify *Dillí* with such features?)

We use the feature "any word" because it allows us to estimate probabilities for words that were never seen as a first word but were seen in another position. If we only had "first word" and "word other than first", then we would not have any probability estimates for the word Dillí when it is encountered as a first word, and we would have to treat it as an unseen word.

f. The first word of each phrase contributes two features for classification (*first word* and *any word*), so in effect it is counted twice. Is this justified? What would happen to *Pan Auto* ("Mr. Auto"), *Nové Marie* ("New Mary"), and *Staré Dillí* ("Old Delhi") if the first word only contributed one feature?

There is more than one way to have the first word contribute only one feature; if we interpret it as contributing the first word feature but not the any word feature, then Pan Auto would now be classified as not a named entity (which is probably a mistake), Nové Marie would be classified as a named entity (same as before), and Staré Dillí would be classified as a named entity (which is probably correct). So based on this limited data we cannot make an argument either for or against counting the first word twice: both methods make errors, though the errors are different.

Problem 4. In this problem you will use probabilities to identify German noun phrases as subjects or objects (a very simple form of *semantic role labeling*). The following sentences are used as training data, with noun phrases annotated as subject or object.

German sentence	English translation
[der Mann]$_{subj}$ sieht [die Frau]$_{obj}$	"The man sees the woman"
[das Kind]$_{subj}$ sieht [den Hund]$_{obj}$	"The child sees the dog"
[den Mann]$_{obj}$ sieht [die Katze]$_{subj}$	"The cat sees the man"
[der Hund]$_{subj}$ sieht [den Mann]$_{obj}$	"The dog sees the man"
[die Frau]$_{subj}$ sieht [die Katze]$_{obj}$	"The woman sees the cat"
[das Kind]$_{subj}$ sieht [die Ziege]$_{obj}$	"The child sees the goat"
[den Hund]$_{obj}$ sieht [der Mann]$_{subj}$	"The man sees the dog"
[der Mann]$_{subj}$ sieht [das Kind]$_{obj}$	"The man sees the child"
[die Katze]$_{obj}$ sieht [das Kind]$_{subj}$	"The child sees the cat"
[das Kind]$_{subj}$ sieht [die Frau]$_{obj}$	"The child sees the woman"

We will use a Naive Bayes classifier to classify the nouns based on two features: the article (*der, die, das, den*) and the position in the sentence (first or second).

a. Identify the priors for each class:

Subject: $\frac{10}{20} = 0.5$ Object: $\frac{10}{20} = 0.5$

b. Give the maximum likelihood estimates for the probability of each feature value, given the class:

	Article				Position	
	der	*die*	*das*	*den*	First	Second
Subject	*0.4*	*0.2*	*0.4*	*0*	*0.7*	*0.3*
Object	*0*	*0.5*	*0.1*	*0.4*	*0.3*	*0.7*

c. Use the classifier to predict the semantic role for each noun phrase in the following sentence: for each noun phrase, calculate the probability that it's generated as subject or object, and then select the most probable class.

[den Mann] sieht [die Frau] "The woman sees the man"

	P(subject)	P(object)	Chosen label
den Mann	$0.5 \times 0 \times 0.7 = 0$	$0.5 \times 0.4 \times 0.3 = 0.06$	*Object*
die Frau	$0.5 \times 0.2 \times 0.3 = 0.03$	$0.5 \times 0.5 \times 0.7 = 0.175$	*Object*

The correct semantic role labels for the above sentence are:

[den Mann]$_{obj}$ sieht [die Frau]$_{subj}$ "The woman sees the man"

The Naive Bayes classifier should have gotten one of the labels wrong (if not, check your math).

d. Why did the classifier make a wrong prediction? What information is missing from the current model?

The classifier labels each noun phrase independently, and [die Frau] in the above sentence is more likely as object (by each feature separately, and by both combined). The model is missing the information that a verb (or sentence) is much more likely to have a subject and an object than two objects and no subject.

Note: conditioning on the nouns won't help, because according to the data, the noun Frau is also more likely as object.

e. Would smoothing on one or both of the features help? What assumptions about the German language would make smoothing desirable or undesirable?

Smoothing will not help get the correct label of [die Frau], but aggressive smoothing on the Article feature could lead to a wrong label on [den Mann], if smoothing causes the difference in the Article feature, which favors object, to be smaller than that of the Position feature, which favors subject.

If we assume that the zero counts in the Article feature reflect general properties of German (der can never be object, den can never be subject), then smoothing is undesirable. If the zero counts might be the result of poor sampling, then smoothing is desirable.

Problem 5. This exercise traces through the first few steps of a perceptron training algorithm. The task is to classify a sentence into one of two classes, which are called $+1$ and -1. We will use just two features; unlike the example in class, these features are not binary, but integer-valued. The features are:

pron The number of personal pronouns in the sentence.

noun The number of proper and common nouns in the sentence.

In the data below, each instance of **pron** is marked in **red boldface**, and each instance of ***noun*** is marked in ***green bold italics***. A hyphenated term such as *Cochin-China* or *great-aunt* is considered a single term. The data (classes and sentences) are taken from Argamon et al.: Gender, genre, and writing style in formal written texts, *Text* **23**(3): 321–346, 2003.

a. Count the features in each sentence and update the perceptron weights, using Algorithm 5 of Hal Daumé III, A Course in Machine Learning (v. 0.99 draft), Chapter 4: The Perceptron.

$+1$ ***Clara*** never failed to be astonished by the extraordinary ***felicity*** of **her** own ***name***.

Feature counts: **pron** ___1___ ***noun*** ___3___ Weights: **pron** ___1___ ***noun*** ___3___ bias ___1___

-1 By 1925 present-day ***Vietnam*** was divided into three ***parts*** under French colonial ***rule***.

Feature counts: **pron** ___0___ ***noun*** ___3___ Weights: **pron** ___1___ ***noun*** ___0___ bias ___0___

$+1$ **She** found it hard to trust **herself** to the mercy of ***fate***, which had managed over the ***years*** to convert **her** greatest ***shame*** into one of **her** greatest ***assets***, and even after ***years*** of comparative ***security*** **she** was still prepared for, still half expecting the old ***gibes*** to be revived.

Feature counts: **pron** ___5___ ***noun*** ___7___ Weights: **pron** ___1___ ***noun*** ___0___ bias ___0___

-1 The southern ***region*** embracing ***Saigon*** and the ***Mekong delta*** was the ***colony*** of ***Cochin-China***; the central ***area*** with its imperial ***capital*** at ***Hue*** was the ***protectorate*** of ***Annam***; and the northern ***region***, ***Tongking***, was also a separate ***protectorate*** with its ***capital*** at ***Hanoi***.

Feature counts: **pron** ___0___ ***noun*** ___16___ Weights: **pron** ___1___ ***noun*** ___−16___ bias ___−1___

$+1$ But whenever **she** was introduced, nothing greeted the amazing, all-revealing ***Clara*** but ***cries*** of "How delightful, how charming, how unusual, how fortunate," and **she** could foresee a ***time*** when ***friends*** would name **their** ***babies*** after **her** and refer back to **her** with ***pride*** as the ***original*** from which ***inspiration*** had first been drawn.

Feature counts: **pron** ___5___ ***noun*** ___8___ Weights: **pron** ___6___ ***noun*** ___−8___ bias ___0___

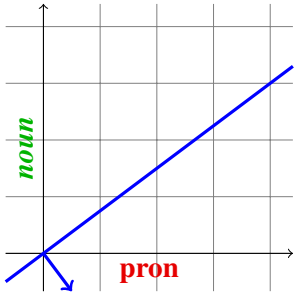
-1 The Annamese ***emperor***, ***Khai Dinh***, in ***theory*** ruled the two northern ***regions*** from ***Hue*** with the ***benefit*** of French ***protection***, while ***Cochin-China*** was governed directly from ***Paris*** but in ***effect*** all three ***territories*** were ruled as ***colonies***.

Feature counts: **pron** ___0___ ***noun*** ___13___ Weights: **pron** ___6___ ***noun*** ___−8___ bias ___0___

b. What is the decision boundary found by the perceptron? Give a formula, and draw the boundary on the graph with a vector pointing in the direction of the positive class (similar to Figures 4.6 and 4.9 in the reading).

Formula: $y = \frac{6}{8}x = 0.75x$

alternative formulations accepted, e.g. $6x - 8y = 0$

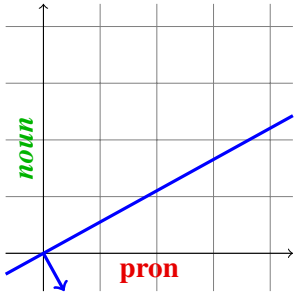


c. Suppose instead of the vanilla perceptron algorithm, we used an *averaged* perceptron (section 4.6 in the reading). What would be the decision boundary? Give a formula and draw the boundary on the graph.

Formula: $y = \frac{16}{29}x \approx 0.552x$

alternative formulations accepted, e.g. $16x - 29y = 0$

Averaging the vectors from (a) gives weights $\frac{16}{6} \approx 2.667$ *and* $\frac{-29}{6} \approx -4.833$*; Algorithm 7 gives weights* $\frac{16}{7} \approx 2.286$ *and* $\frac{-29}{7} \approx -4.143$*. These define the same decision boundary!*



d. How would each of the perceptrons (vanilla and averaged) classify each of the following texts?

Finally **her** ***confidence*** grew to such an ***extent*** that **she** was able to explain that **she** had been christened not in the ***vanguard*** but in the extreme ***rearguard*** of ***fashion***, after a Wesleyan ***great-aunt***, and that **her** ***mother*** had formed the ***notion*** not as an unusual and charming ***conceit*** but as a preconceived ***penance*** for **her** ***daughter***, whose only ***offences*** at that tender ***age*** were **her** ***existence*** and **her** ***sex***.

Vanilla: -1 Averaged: -1

Some backward ***tribes*** inhabited the remoter ***mountains*** and ***jungles*** but the main ***population*** was of the same ***race***; today **they** are known as ***Vietnamese*** but then the outside ***world*** knew **them** as ***Annamites*** or ***Annamese***.

Vanilla: -1 Averaged: -1

Problem 6. Trace the first steps of training a perceptron to classify a tweet into one of two classes, which are called $+1$ and -1. The perceptron uses just two integer-valued features:

- sent.: The number of sentences in the tweet consisting of a single word.
- CAPS: The number of words in ALL CAPS.

The perceptron training algorithm is given below: w_d is the weight of feature d, x_d is the count of feature d in a particular item, and *class* is $+1$ or -1. (Modeled on the algorithm in the reading: Hal Daumé III, A Course in Machine Learning (v. 0.99 draft), Chapter 4: The Perceptron)

```
w_d ← 0 for all features d                          # Initialize weights
b ← 0                                               # Initialize bias
for all iterations:
    for all items:
        activation ← Σ_d w_d x_d + b                #Compute activation
        if class · activation ≤ 0 :
            w_d ← w_d + class · x_d for all features d   # update weights
            b ← b + class                           # update bias
return w_1, w_2, ..., b
```

a. Count the features in each sentence and update the perceptron weights.

$+1$ Epic crowds in Pennsylvania tonight, but FAKE MEDIA won't report it. SAD.

Feature counts: sent. 1 CAPS 3 Weights: sent. 1 CAPS 3 bias 1

-1 High volatility in NASDAQ before closing with moderate gains.

Feature counts: sent. 0 CAPS 1 Weights: sent. 1 CAPS 2 bias 0

$+1$ Senate must choose tonight to protect America. Every vote counts!

Feature counts: sent. 0 CAPS 0 Weights: sent. 1 CAPS 2 bias 1

-1 Immediate release. Fed to raise interest for third month in a row.

Feature counts: sent. 0 CAPS 0 Weights: sent. 1 CAPS 2 bias 0

$+1$ It's wonderful to see the effect of our TAX CUTS. Phenomenal.

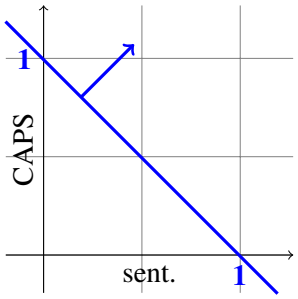
Feature counts: sent. 1 CAPS 2 Weights: sent. 1 CAPS 2 bias 0

-1 NYSE opening to a hesitant start.

Feature counts: sent. 0 CAPS 1 Weights: sent. 1 CAPS 1 bias -1

b. What is the decision boundary found by the perceptron? Give a formula, and draw the boundary on the graph with a vector pointing in the direction of the positive class.

Formula: $y = 1 - x$
Alternatives accepted, e.g. $x + y - 1 = 0$
Exact location of graph can vary (since the grids are unlabeled), but the slope should be reasonably accurate and it should be clear where the boundary crosses the axes.



c. How would the perceptron classify each of the following tweets?

Please join me with your thoughts and prayers for both aviators, their families and our incredible @USNavy.

Class: -1

Today the House took major steps toward securing our schools by passing the STOP School Violence Act. We must put the safety of America's children FIRST.

Class: $+1$

We cannot keep a blind eye to the rampant unfair trade practices against our Country!

Class: -1

Unemployment filings are at their lowest level in over 48 years. Great news for workers and JOBS, JOBS, JOBS!

Class: $+1$

Problem 7. We have seen that in a naive Bayes model with features $f_1, f_2 \ldots$, for a specific text with corresponding feature counts $n_1, n_2 \ldots$, the log probability that the text belongs in a particular class is given by the model as follows:

$$\log P(class|text) \approx \log P(class) + n_1 \log P(f_1|class) + n_2 \log P(f_2|class) + \cdots$$

That is, the log probability of class membership is proportional to the distance above a plane corresponding to the class. The normal to the plane is a weight vector $\mathbf{w} = w_1, w_2 \ldots$ where for all features f_i, $w_i = \log P(f_i|class)$. (We can consider the log prior probability $\log P(class)$ as an extra feature w_0 where for all texts, $n_0 = 1$.)

For the following parts, assume we have two classes C_1 and C_2, with associated weight vectors $\mathbf{w}_1$ and $\mathbf{w}_2$.

a. Given a text represented by a feature count vector $\mathbf{n} = n_1, n_2 \ldots$, when will the model classify the text as belonging to class C_1? When will the model classify the text as belonging to class C_2? Give the answers in terms of $\mathbf{w}_1$, $\mathbf{w}_2$ and $\mathbf{n}$.

The model will choose class C_1 if $\mathbf{w}_1 \cdot \mathbf{n} > \mathbf{w}_2 \cdot \mathbf{n}$
The model will choose class C_2 if $\mathbf{w}_1 \cdot \mathbf{n} < \mathbf{w}_2 \cdot \mathbf{n}$

Alternative formulations acceptable, e.g. $C_1 : \mathbf{w}_1 \cdot \mathbf{n} - \mathbf{w}_2 \cdot \mathbf{n} > 0$
Including priors in the formula is acceptable but not necessary (since the problem states that priors may be considered part of the vector)

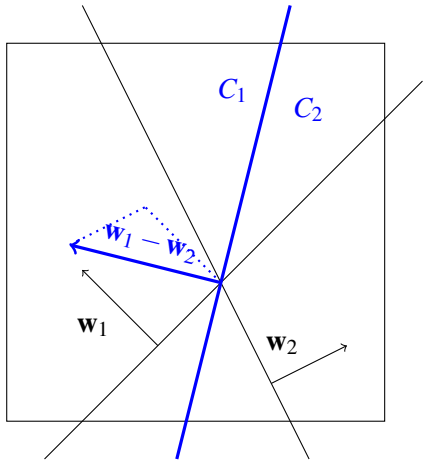
b. Given your answer above, how can we represent the *decision boundary* between the classes C_1 and C_2? In which direction from the boundary are texts classified as C_1, and in which direction as C_2? Give the answers in terms of $\mathbf{w}_1$ and $\mathbf{w}_2$.

The decision boundary is the plane perpendicular to the vector $\mathbf{w}_1 - \mathbf{w}_2$ (or $\mathbf{w}_2 - \mathbf{w}_1$), offset from the origin by the difference in log prior probabilities
The vector $\mathbf{w}_1 - \mathbf{w}_2$ points in the direction of C_1, while $\mathbf{w}_2 - \mathbf{w}_1$ points in the direction of C_2

Alternative formulations acceptable, e.g. the plane that consists of all points $\mathbf{x}$ such that $(\mathbf{w}_1 - \mathbf{w}_2) \cdot \mathbf{x} = 0$
Including priors in the formula is acceptable but not necessary (since the problem states that priors may be considered part of the vector)

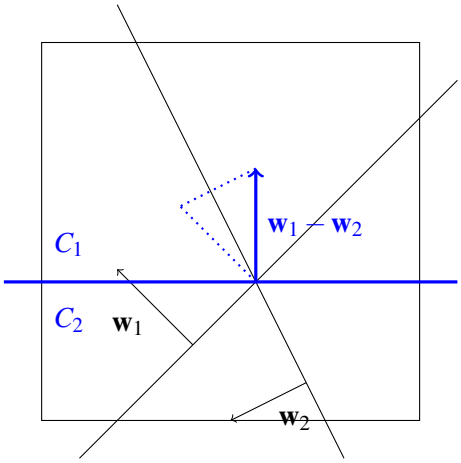
c. The following box represents a 2-dimensional feature space, with the planes and weight vectors associated with C_1 and C_2. Use a diagram and an explanatory sentence to show how these planes determine a decision boundary, and indicate the decision regions (that is, which part of the feature space will be classified as C_1 and which as C_2).

The boundary is the plane perpendicular to the difference of $\mathbf{w}_1$ *and* $\mathbf{w}_2$
Alternative formulations accepted, e.g. a plane of points whose distance from the two planes is weighted by $\mathbf{w}_1$ *and* $\mathbf{w}_2$

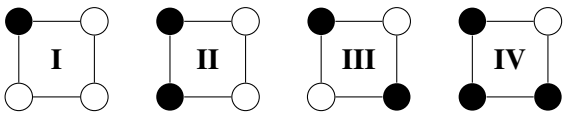


d. Do the same for the following case (diagram and explanatory sentence). What is the difference?

The difference from the previous problem is in the direction of $\mathbf{w}_2$

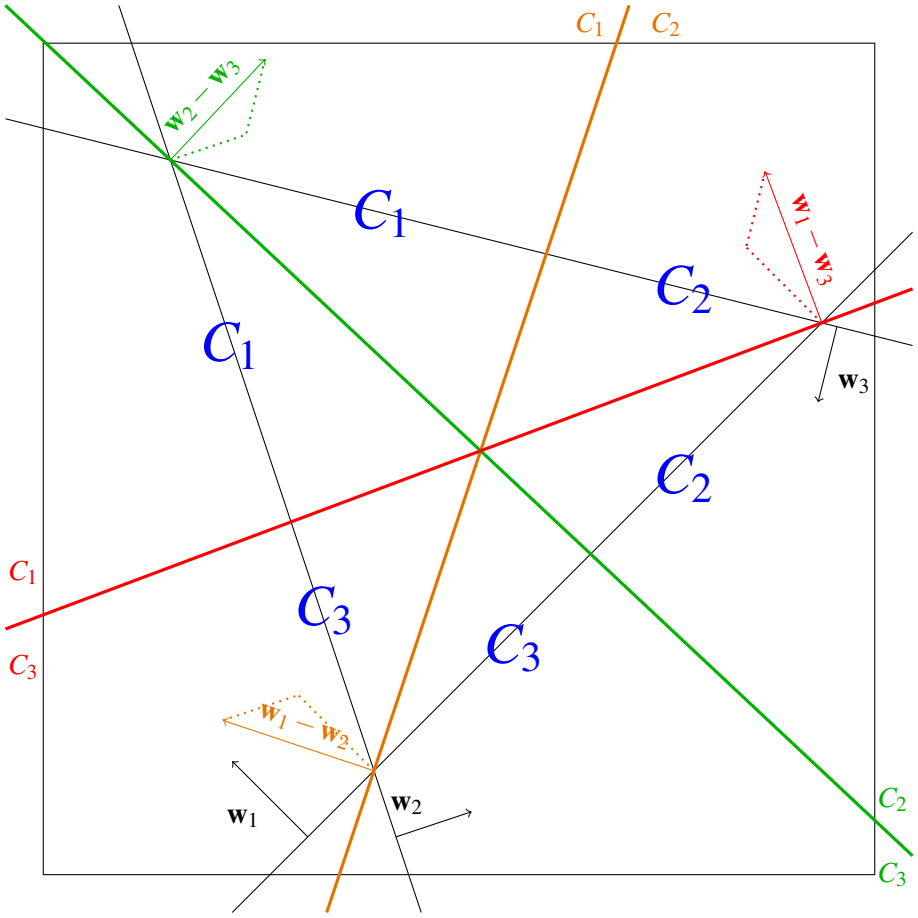


e. The following diagrams represent possible ways to split the four possible observations of two binary features between two classes. Which of the cases below are consistent with conditional independence of the feature values, given the class?



Answer: ***I, II*** *and* ***IV***

Problem 8. The following box represents a 2-dimensional feature space (in log space), with the planes and weight vectors associated with a three-class naive Bayes classifier. Classes C_1, C_2 and C_3 are associated with weight vectors $\mathbf{w}_1$, $\mathbf{w}_2$ and $\mathbf{w}_3$, respectively. Here, every *pair* of classes determines a decision boundary.



a. On the diagram, draw and label the decision boundary for each pair of classes. That is, draw the decision boundary between C_1 and C_2, and label which class is on each side of the boundary. Do the same for the boundary between C_1 and C_3, and for the boundary between C_2 and C_3.

Labeled in the diagram

b. On the diagram, identify and label the decision regions for each class. That is, which parts of the space will be classified as C_1, which as C_2, and which as C_3.

Labeled in the diagram

Note: Regions are the determined by the decision boundaries (above in thick colors), not the original planes (thin black lines).