10-601: Homework 2

Due: 25 September 2014 11:59pm (Autolab)

TAs: Siddhartha Jain, Ying Yang

Name: Matthew Tay

Andrew ID: mhtay

Please answer to the point, and do not spend time/space giving irrelevant details. Please state any additional assumptions you make while answering the questions. For Questions 1 to 5, 6(b) and 6(c), you need to submit your answers in a single PDF file on autolab, either a scanned handwritten version or a LaTeXpdf file. Please make sure you write legibly for grading. For Question 6(a), submit your m-files on autolab.

You can work in groups. However, no written notes can be shared, or taken during group discussions. You may ask clarifying questions on Piazza. However, under no circumstances should you reveal any part of the answer publicly on Piazza or any other public website. The intention of this policy is to facilitate learning, not circumvent it. Any incidents of plagiarism will be handled in accordance with CMU's Policy on Academic Integrity.

⋆: Code of Conduct Declaration

- Did you receive any help whatsoever from anyone in solving this assignment? Yes / No.
- If you answered *yes*, give full details: _____ (e.g. *Jane explained to me what is asked in Question 3.4*)
- Did you give any help whatsoever to anyone in solving this assignment? Yes / No.
- If you answered *yes*, give full details: _____ (e.g. *I pointed Joe to section 2.3 to help him with Question 2*).

1: A probabilistic view of linear regression. (TA:- Ying Yang)

Let X and Y be two random variables, β be a constant vector, and $\epsilon \sim \mathcal{N}(0, \sigma^2)$ be a Gaussian random variable with zero mean and variance σ^2. We assume $Y = \beta X + \epsilon$, and that ϵ is independent of X.

(a) Show that given $X = x$, the distribution of Y is $\mathcal{N}(\beta x, \sigma^2)$ *[3 points]*

$E(Y) = E(\beta X + \epsilon) = E(\beta X) + E(\epsilon)$

$= \beta E(X) + E(\epsilon)$

$= \beta x + 0$ (since $X = x$, & $\mu_\epsilon = 0$)

$Var(Y) = E((Y - E(Y))^2) = E((\beta X + \epsilon - \beta x)^2)$

$= E(\epsilon^2)$

$Var(\epsilon) = \sigma^2 \Rightarrow E(\epsilon^2) - E(\epsilon)^2 = \sigma^2 \Rightarrow E(\epsilon^2) = \sigma^2 + E(\epsilon)$

$= \sigma^2 \Rightarrow Y \sim N(\beta x, \sigma^2)$

qed

(b) Let $\{(X_i, Y_i), i = 1, \cdots, n\}$ be n independent samples from the model above. Show that the maximum likelihood estimation of β, where the likelihood is with regard to the conditional distribution $Y|X$, is the least square solution

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - \beta X_i)^2$$

[3 points]

$\max L(\beta) = \arg\max_{\beta} \prod_i P(Y_i \mid X_i, \beta)$

$= \arg\max_{\beta} \prod_i \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y_i - \beta X_i)^2}{2\sigma^2}}$

$\ln L(\beta) = \ln \left(\prod_i \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y_i - \beta X_i)^2}{2\sigma^2}} \right)$

$= \sum_i \ln \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y_i - \beta X_i)^2}{2\sigma^2}}$

$= \sum_i \ln \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_i (Y_i - \beta X_i)^2$ ↑ want to be minimized for max $\ln L(\beta)$

$\arg\max_{\beta} \ln L(\beta) \Rightarrow \min \frac{1}{2\sigma^2} \sum_i (Y_i - \beta X_i)^2$

$\Rightarrow \min \sum_i (Y_i - \beta X_i)^2$

$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{n} (Y_i - \beta X_i)^2$

qed

2: One-dimensional ridge regression(TA:- Ying Yang)

Let Y and X be two random variables, and $Y = \beta X + \epsilon$ given X, where β is a constant, and $\epsilon \sim \mathcal{N}(0, \sigma^2)$, independent of X. Given n independent sample pairs, $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$, instead of ordinary least square, here we estimate β with "ridge regression", by solving the following problem.

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{2} \left(\sum_{i=1}^{n} (y_i - \beta x_i)^2 + \lambda \beta^2 \right)$$

where $\lambda \geq 0$ is a tuning parameter.

(a)Give a solution in explicit formula for $\hat{\beta}$. [3 points]

$\frac{\partial}{\partial \beta}\left(\frac{1}{2}\left(\sum_{i=1}^{n}(y_i - \beta x_i)^2 + \lambda\beta^2\right)\right) = 0$ (at min)

$\frac{1}{2} \cdot \sum_i 2 \cdot (y_i - \beta x_i) \cdot (-x_i) + \frac{1}{2} 2\lambda\beta$

$= -\sum_i (y_i - \beta x_i) x_i + \lambda\beta = 0$

$\lambda\beta = \sum_i (y_i - \beta x_i)$

$\lambda\beta = \sum_i y_i - \beta \sum_i x_i$

$\beta(\lambda + \sum_i x_i) = \sum_i y_i$

$\hat{\beta} = \frac{\sum_i y_i}{\lambda + \sum_i x_i}$ (at min $\hat{\beta} = \beta$)

(b) When λ goes from 0 to infinity, how does $\hat{\beta}$ change? Give a brief explanation of your answer. [2 points]

At $\lambda = 0$, no regularization occurs. $\hat{\beta}$ estimate purely from maximum likelihood estimate (i.e from the data).

As λ increases to infinity, regularization occurs. $\hat{\beta}$ is reduced by factor of $\frac{\sum_i x_i}{\lambda + \sum_i x_i}$. This means that the linear model that fits the data has its parameters β smaller than by pure observation of the data; to prevent over fitting usually.

At $\lambda =$ infinity, $\hat{\beta} = 0$, and all information from data is ignored.

3: Least square (TA:- Ying Yang)

Suppose X and Y are random variables. Let $(x_1, y_1), \cdots, (x_n, y_n)$ be n pairs of samples. Compute the least square solutions for the following models. $\epsilon \sim N(0, \sigma^2)$

1. $Y = \beta X + \epsilon$
2. $Y = \beta^2 X + \epsilon$

Which of the the models above yields to a lower training error?

[5 points]

1. ref to Q1(b), least squares solution:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n} (Y_i - \beta X_i)^2$$

$$\frac{\partial}{\partial \beta} \sum_i (Y_i - \beta X_i)^2 = 2 \sum_i (Y_i - \beta X_i)(-X_i)$$

$$= -2 \sum_i (Y_i - \beta X_i) X_i$$

$$= 0 \quad \text{(at min)}$$

$$\sum_i (Y_i - \beta X_i) X_i = 0$$

$$\sum_i Y_i X_i - \sum_i \beta (X_i^2) = 0$$

$$\beta \sum_i X_i^2 = \sum_i Y_i X_i$$

$$\hat{\beta} = \frac{\sum_i Y_i X_i}{\sum_i X_i^2} \quad (\hat{\beta} = \beta \text{ at min})$$

2. $\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \sum_i (Y_i - \beta^2 X_i)^2$

$$\frac{\partial}{\partial \beta} \sum_i (Y_i - \beta^2 X_i)^2 = 2 \sum_i (Y_i - \beta^2 X_i) \cdot (-2\beta X_i) = 0 \quad \text{(at min)}$$

$$\sum_i Y_i X_i - \sum_i \beta^2 (X_i)^2 = 0$$

$$\beta^2 \sum_i (X_i)^2 = \sum_i Y_i X_i$$

$$\hat{\beta} = \pm \sqrt{\frac{\sum_i Y_i X_i}{\sum_i X_i^2}}$$

Model 1 has lower training error. This assumes that in calculation of sqrt(expr), some precision is lost in finding $\hat{\beta}$ for model 2. Also predictor Y such that constant term $\hat{\beta}^2$ always positive when minimizing w.r.t β $\Rightarrow$ some hypothesis not covered, and true $\beta \geq 0$, Otherwise, if no precision lost, both models equally accurate as they are essentially the same model.

4: Behavior of linear regression (TA:- Siddhartha Jain)

Suppose you know the number of keyboard and mice sold at various locations around the world and from that you want to estimate the number of computers sold using linear regression. Your model is $Y = \beta_1 k + \beta_2 m$ where Y is the number of computers sold, k is the number of keyboards sold and m is the number of mice sold. You get 101 observations such that 100 of them have 1 keyboard, 1 mouse and 1 computer, but the 101st has 1 keyboard, 0 mouse, and 1 computer.

For **(a)** and **(b)**, you can use `regress` in Matlab to compute the answers.

(a) What are the optimal values of β_1, β_2 in the scenario above.

Matlab code:

```
m = ones(100,1);
m = cat(1,m,0);
k = ones(101,1);
Y = ones(101,1);
X = cat(2,k,m);
B = regress(Y,X);
```

$\beta_1 = 1.000$

$\beta_2 = -1.4505 \times 10^{-15}$ #

[3 points]

(b) Now suppose you get two additional observations, both with 0 keyboard, 1 mouse, and 1 computer. What are the optimal β values now?

```
m = cat(1,m,ones(2,1));
k = cat(1,k,zeros(2,1));
X = cat(2,k,m);
Y = ones(103,1);
B = regress(Y,X);
```

$\beta_1 = 0.3377$

$\beta_2 = 0.6689$ #

[3 points]

(c) As you should notice, the optimal values for β fluctuate wildly with the addition of even very few observations. This is a problem as then it's hard to converge on a set of values for β. Why is this behavior happening? Given an arbitrary dataset X, Y, how can we test whether such behavior might occur?

Linear regression minimizes the least squares distance (l_2 norm) between the model's prediction and the labels observed from the data. [3 points]
Thus it is vulnerable to outliers, like in part (b), where suddenly 2 points that have 0 keyboards have 1 computer while all 101 examples before did have 1 keyboard when 1 computer appeared. Because the distance of the outliers is great, the model changes drastically to adapt to every new outlier point, thus β changes much.
To test for such behavior, one can compute the σ^2 (or σ) for each predictor variable. If there are outliers outside 2 standard deviations, such fluctuating behavior in β might occur. 5

5: Gaussian Naive Bayes. (TA:- Ying Yang)

Let $Y \in \{0, 1\}$ be class labels, and let $X \in \mathbb{R}^p$ denote a p-dimensional feature.

(a) In a Gaussian naive Bayes model, where the conditional distribution of each feature is a one-dimensional Gaussian, give a maximum-likelihood estimate (MLE) of the conditional distribution of feature $X^{(j)}, j = 1, \cdots, p$, $(X^{(j)}|Y \sim N(\mu_Y^{(j)}, (\sigma_Y^{(j)})^2))$ *[4 points]*

$L(\mu_x^j, \sigma_x^j) = \prod_i P(Y \mid X^{(j)} \sim N(\mu_x^{(j)}, (\sigma_x^{(j)})^2))$

$\underset{\mu_x^j, \sigma_x^j}{\text{argmax}} \ln L(\mu_x^j, \sigma_x^j) = \underset{\mu_x^j, \sigma_x^j}{\text{argmax}} \ln \prod_i \frac{1}{\sqrt{2\pi}\sigma_x^j} e^{-\frac{(x-\mu_x^{(j)})^2}{2(\sigma_x^j)^2}}$

$= \underset{\mu, \sigma}{\text{argmax}} \sum_i \ln \frac{1}{\sqrt{2\pi}\sigma_x^j} - \frac{1}{2(\sigma_x^j)^2} \sum_i (x - \mu_x^{(j)})^2$

(differentiating) maximizing w.r.t μ_x^j, σ_x^j: $\left(-\frac{1}{(\sigma_x^j)^2} \sum_i (x - \mu_x^j)\right) = 0$ $\quad \frac{\partial}{\partial \sigma_x^j}\left[\sum_i \left(-\ln(\sqrt{2\pi}\sigma_x^j)\right) - \frac{1}{2(\sigma_x^j)^2} \sum_i (x-\mu_x^j)^2\right] = 0$

$-\frac{1}{(\sigma_x^j)^2} \sum_i x_i + \frac{1}{(\sigma_x^j)^2} \sum_i \mu_x^j = 0$ $\quad -\frac{\sqrt{2\pi} \cdot n}{\sqrt{2\pi}\sigma_x^j} + \frac{1}{(\sigma_x^j)^3} \sum_i (x-\mu_x^j)^2 = 0$

From working on right: $X^{(j)} \sim N\left(\frac{\sum_i x_i^j}{n}, \frac{\sum_i (x_i - \mu_x^j)^2}{n}\right)$ $\quad \sum_i x_i = \sum_i \mu_x^j = n\mu_x^j$ $\quad n = \frac{1}{(\sigma_x^j)^2} \sum_i (x_i - \mu_x^j)^2$

$\mu_x^j = \frac{\sum_i x_i^j}{n}$ $\quad (\sigma_x^j)^2 = \frac{\sum_i (x_i^j - \mu_x^j)^2}{n}$ ✱ ✱

(b) In a full Gaussian Bayes model, we assume that the conditional distribution $\Pr(X|Y)$ is a multidimensional Gaussian, $X|Y \sim \mathcal{N}(\mu_Y, \Sigma_Y)$, where μ is the mean vector and $\Sigma \in \mathbb{R}^{p \times p}$ is the covariance matrix. Suppose the prior of Y is already given. How many parameters do you need to estimate in Gaussian naive Bayes model? How many in a full Gaussian Bayes model? *[3 points]*

Gaussian naive Bayes: Assume conditional independence

$\Rightarrow P(X_i^1, X_i^2 \mid Y) = P(X_i^1 \mid Y) \cdot P(X_i^2 \mid Y)$

num parameters to estimate: p times (of μ_x^j) + p times (of $(\sigma_x^j)^2$)

$= 2p$ ✱

full Gaussian Bayes: Does not assume ✱ conditional independence. Correlation possible between X^j, X^i

num parameters to estimate: p times (of μ_x^j) + p^2 (every entry of covariance matrix Σ $\sigma_{x_{ij}}$ $\forall i, j \in \{1, 2, 3, \cdots p\}$)

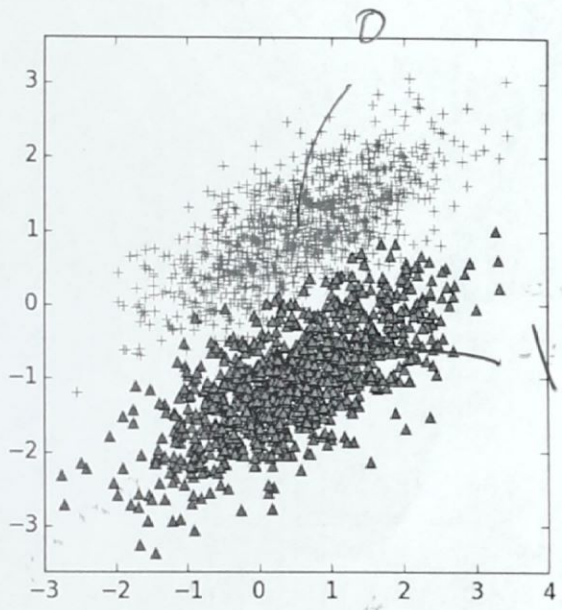
$= p^2 + p$ ✱

(c) In a two dimensional case, we can visualize how Naive Bayes behaves when input features are correlated. A data set shown in Figure 1 (A), where red points are in Class 0, blue points are in Class 1. The conditional distributions are two-dimensional Gaussians. In (B) (C) and (D), the ellipses represent conditional distributions for each class. The centers of ellipses show the mean and the contours show the boundary of two standard deviations. Which of them is most likely to be the true conditional distribution? Which of them is most likely to be estimates by a Gaussian naive Bayes model? If we assume the prior probabilities for both classes are equal, which model will achieve a higher accuracy on the training data? *[3 points]*

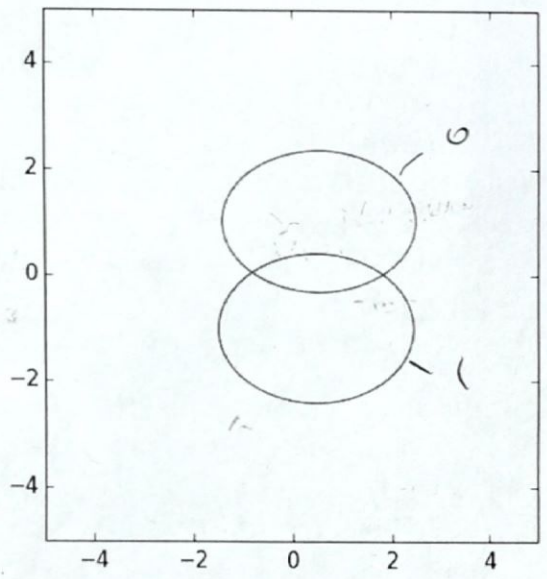
B - true conditional distribution; slight elliptical shape showing ✱ points within 2 standard deviations cover greater range in 1 dimension than the other. The overlap reflects correlation between X_1, X_2 observed in the data (when X_1 increases, X_2 increases / slope)

C - no overlap $P(X_1, X_2 \mid Y) = P(X_1 \mid Y) P(X_2 \mid Y)$ Estimates from Gaussian Naive Bayes model. ✱

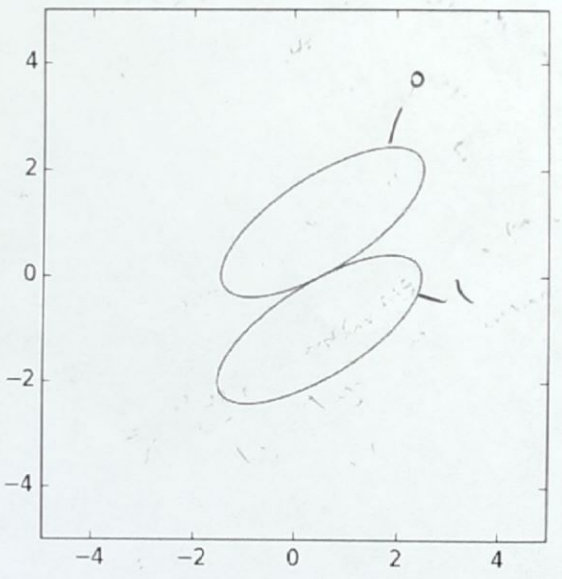
B - higher accuracy; assuming equal priors, B reflects correlation found in data.



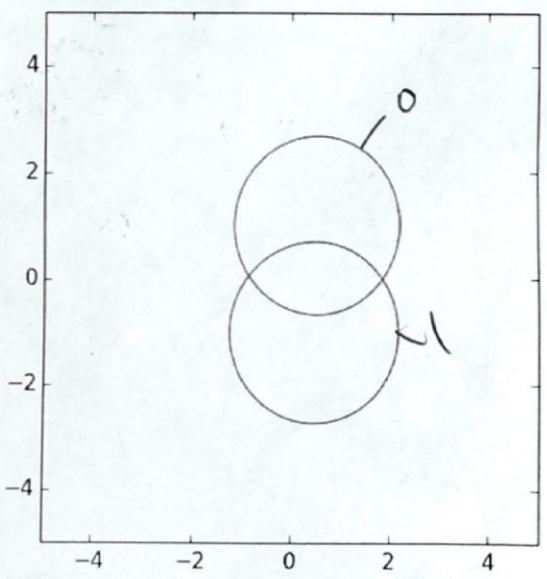
(A) Data



(B)



(C)



(D)

Figure 1: Figure of Q6 (c)

6: Text classification using Naive Bayes. (TA:- Siddhartha Jain& Ying Yang)

In this assignment, you are going to program a naive Bayes classifier to classify documents from a serious European magazine "economist" (Class 1) and a not-so-serious American megazine "the onion" (Class 0).

1. **Data description**
   All data files are in `Onion_vs_Economist`. If you load the `handout.mat` into Octave (or Matlab) with `load handout.mat`, you will see the following matrices, `Xtrain, Ytrain, Xtest, Ytest`. We also provided a dictionary of V tokens (or words) in `dictionary.mat`, and denote the tokens in the dictionary by indices, $\{1, 2, \cdots, V\}$. There are n training documents and m testing documents. For each document, we counted the number of occurrence of each token, resulting in a vector $(c_1, c_2, \cdots, c_V)$. Each row in `Xtrain` and `Xtrain` is such a vector for one document. `Ytrain` and `Ytest` are $n \times 1$ and $m \times 1$ binary class labels.

2. **Model description (multinomial model)**
   We view a document as an ordered sequence of word events. Suppose we have a document with label $Y = y \in \{0, 1\}$, which contains q words in total, we use the event $W_i = j$ to denote the event that the ith word is the jth token in the dictionary, $j \in \{1, 2, \cdots, V\}$. With a naive Bayes model, we assume that the q word events are independent, and have an identical multinomial distribution with V outcomes.

   Learning the conditional probability
   Given one training document in Class y, if we do not use smoothing (or pseudocounts), we estimate the conditional probability for a word event W in the following way,

$$\begin{aligned}\Pr(W = j|Y = y) &= \frac{\text{number of occurance of token } j}{\text{total number of words}} \\ &= \frac{\text{number of occurance of token } j}{\text{total number of occurance of all } V \text{ tokens}}\end{aligned}$$

   In `Xtrain`, you are given multiple training documents in one class, you should think in a way as concatenating them all into a large document. You need to use additive smoothing (or pseudocount) http://en.wikipedia.org/wiki/Additive_smoothing in your implementation, setting $\alpha = 1$.

   Learning the prior
   Assume the prior distribution of label Y is binomial, without smoothing, it is estimated as

$$\Pr(Y = y) = \frac{\text{number documents in Class y}}{\text{total number of documents}}$$

   Making prediction
   Now given the test document of length q,

$$y* = \arg\max_y \Pr(Y = y|W_1, \cdots, W_q) = \arg\max_y \frac{\prod_{i=1}^{q} \Pr(W_i|Y = y) \Pr(Y = y)}{\Pr(W_1, \cdots, W_q)}$$

$$= \arg\max_y (\prod_{i=1}^{q} \Pr(W_i|Y = y) \Pr(Y = y))$$

However, we are only given the word counts of the document, $(c_1, c_2, \cdots, c_V)$, and we can only compute the multinomial probability.

$$y* = \arg\max_y (q! \prod_{j=1}^{V} \frac{\Pr(W = j|Y = y)^{c_j}}{c_j!} \Pr(Y = y) \quad (1)$$

$$= \arg\max_y (\sum_{j=1}^{V} c_j \log \Pr(W = j|Y = y) + \log \Pr(Y = y)) + \log(q!) - \sum_{j=1}^{V} \log(c_j!) \quad (2)$$

$$= \arg\max_y (\sum_{j=1}^{V} c_j \log \Pr(W = j|Y = y) + \log \Pr(Y = y)) + \text{constant} \quad (3)$$

In your implementation, to avoid multiplying very small probabilities and underflow, you should use the logarithmic transformation as in Equation 3.

For **(a)** submit your m-files to autolab. For **(b)** and **(c)**, write your solutions in your pdf.

(a) Create following three octave functions and save them in three files, `nb_train.m`, `nb_test.m` and `nb_run.m`.

```
model = nb_train(Xtrain, Y_train)
Pred_nb = nb_test(model, Xtest)
accuracy = nb_run(Xtrain, Ytrain, Xtest, Ytest)
```

`model` is a structure that describe the model you learned. `Pred_nb` is a $m \times 1$ binary vector, which denotes your prediction for the testing documents. In `nb_run`, return the prediction accuracy computed by `accuracy = mean(Pred_nb==Ytest)` , and use `save('Pred_nb.mat','Pred_nb')` to save your prediction into a mat file.

Note: Your score will be determined by your classification accuracy on the test dataset you've been given as well as the held-out dataset that has not been released.

[15 points]

(b) For the jth token in the dictionary, we can compute the following log-ratio,

$$\left| \log \frac{Pr(W = j|Y = 1)}{Pr(W = j|Y = 0)} \right|$$

Use this log-ratio as a measure, find the top five words that are most discriminative of the classes, report them in your pdf.

Matlab code:

```
lg_ratio = log(P(w|Y=1))./log(P(w|Y=0));
[sort_r, idx] = sort(lg_ratio);
indices_largest = idx(1:5);
```

Using indices_largest to index into dictionary, top 5 words:

Percent, monday, yankees, sox, schmuck

[5 points]

(c) State the Naive Bayes assumption. Are there any pairs of words that violate the Naive Bayes assumption? If so, give 1 example of such pairs and explain why they might be violating the Naive Bayes assumption.

Naive Bayes assumes that the conditional probabilities for each feature given a class are all independent. $\Rightarrow P(w_1, w_2, \cdots w_n | Y=y_i)$

$= P(w_1|Y=y_i) P(w_2|Y=y_i) \cdots P(w_n|Y=y_i)$.

Example: Percent, Monday. $P(w_{percent}, w_{monday}|Y=0) = \frac{\min\{50,77\}}{(92136)}$ $\left(\frac{\#\text{ of occurances together}}{\#\text{ of words union}}\right) = 5.43 \times 10^{-4}$

$P(w_{percent}|Y=0) P(w_{mon}|Y=0) = \left(\frac{50}{92136}\right)\left(\frac{77}{92136}\right) = 4.54 \times 10^{-7}$

[5 points]

Total: 60

The actual probability violates Naive Bayes assumption because 'percent' is correlated with 'Monday''s occurance. Thus the probability of observing 'Monday' given that 'percent' is observed is higher than the individual conditional probabilities would suggest.

Note: $P(W=\text{percent}, W=\text{monday} \mid Y=0) \geq \frac{\min\{50,77\}}{92136}$

as each training document is independent.