DS-GA 3001, Introduction to Data Science (PhD)
Prof. Arthur Spirling, Prof. Julia Kempe
Assignment date: November 25, 2020, 2020

Homework 3

This homework must be returned to your TA by **5pm Eastern Time, December 5th, 2020**. Late work will incur penalties of the equivalent of one third of a letter grade per day late.

It must be your own work, and your own work only—you must not copy anyone's work, or allow anyone to copy yours. This extends to writing code. You may consult with others, but when you write up, you must do so alone.

Your homework submission must be in one of the following formats: (1) A set of answers and a clearly commented `Python` code appendix (use comments to identify code relevant to each answer you produced), (2) A report consisting of clearly marked answers, each accompanied by the relevant code (e.g., a report generated using Python markdown or similar). **In either case, your code must be included in full, such that your understanding of the problems can be assessed.**

---

1. In this exercise, you will investigate entropy and derive some of its fundamental properties. We will only focus on the discrete case (Shannon entropy), i.e. $H(p) = -\sum_{i=1}^{n} p_i \log p_i$ where $p = (p_1, p_2, \ldots, p_n)$ is some discrete probability distribution so that $p_i \geq 0$ and $\sum_{i=1}^{n} p_i = 1$. Throughout this exercise, you may assume $0 \log 0 = 0$. You may also use concavity of $-x \log x$ where $x \geq 0$ without proof.

   (a) Show that $H(p) \geq 0$. Give an example of a distribution p_0 such that $H(p_0) = 0$.

   (b) An n-variate scalar function f is *concave* if, for all $x, y \in \text{dom}(f)$ and $\lambda \in [0, 1]$, we have $f(\lambda x + (1-\lambda)y) \geq \lambda f(x) + (1-\lambda) f(y)$. Prove that $H(p)$ is concave when viewed as an n-variate function of $p_1, p_2, \ldots, p_n$.

   (c) Prove that if f is concave, then, for any set of n values $\lambda_i \geq 0$ with $\sum_{i=1}^{n} \lambda_i = 1$ and any $x_1, x_2, \ldots, x_n$ from the domain of f, we have $f\left(\sum_{i=1}^{n} \lambda_i x_i\right) \geq \sum_{i=1}^{n} \lambda_i f(x_i)$. (Hint: use induction on n). This result is called Jensen's inequality.

   (d) Use Jensen's inequality to prove that $H(p) \leq \log n$. For which distribution p will $H(p) = \log n$?

2. Noisy coding theorem: In this exercise you will complete the proof of the noisy coding theorem seen in the Lecture. Namely, assume we have a source of iid bits b with probability $\Pr[b = 1] = p$, and without loss of generality $p < \frac{1}{2}$. We want to show:

   Theorem: For all $p < \frac{1}{2}$ and $\epsilon, \delta > 0$ and large enough n, there exists an encoder $E : \{0,1\}^n \to \{0,1\}^m$ and decoder $D : \{0,1\}^m \to \{0,1\}^n$ such that $m \leq n \cdot (H(x) + \epsilon)$ and $Pr_x[D(E(x)) \neq x] < \delta$. Recall

   - Stirling's formula: $n! \approx \sqrt{2\pi n} \left(\frac{n}{e}\right)^n$

- Chernoff bound: Let $X_1, \ldots, X_n$ be iid random variables in $[0, 1]$ with expectation p. Then
$$\Pr\left[\left|\frac{1}{n}\sum_i X_i - p\right| \geq \lambda\right] \leq 2e^{-n\lambda^2}$$
- Define $wt(x) = \#1$ in x. Define a set $A = \{x \in \{0,1\}^n | (p-\lambda)n \leq wt(x) \leq (p+\lambda)n\}$ for some λ such that $\lambda \to 0$ for large n and that you will determine in (b).

(a) Show that A is "small": $\log|A| \leq n(H(p) + \epsilon')$, where ϵ' is a function of λ and $\epsilon' \to 0$ as n grows large.

(b) A randomly drawn x (according to the source distribution) is $\in A$ with high probability: Use the Chernoff bound to show that $\Pr[x \notin A] < \delta$ for a choice of λ.

(c) Combine (a) and (b) to show that for large enough n, $\log|A| \leq n(H(p) + \epsilon)$ and finish the proof of the theorem by using an encoding that gives an index in A.

3. SOM and clustering: Show (as a brief outline only, no formulas or calculations needed) how a SOM becomes a type of k-means clustering when the width of the neighborhood is set to zero. Note that the difference to conventional k-means clustering will be that training data points are added one at a time for each update, as opposed to being present in its entirety at the beginning.

4. VC-dimension:

(a) *Threshold* and *Interval* classifiers: Compute the VC dimension of (1) the class of threshold classifiers $T = \{t_a(x) = I_{x<a} | a \in \mathbb{R}\}$ and (2) the class of interval classifiers $H = \{h_{ab}(x) = I_{x\in(a,b)} | a, b \in \mathbb{R}, a < b\}$

(b) Show that the set of functions $\{I(\sin(\alpha x) > 0)\}$ can shatter the following points on the line:
$$z_1 = 10^{-1}, \ldots, z^l = 10^{-l}$$
for any l. Hence the VC dimension of the class $\{I(\sin(\alpha x) > 0)\}$ is infinite.

5. Gaussian Naive Bayes: in Lab10, we derived the maximum likelihood estimators for Bernoulli and Multinomial conditional models; in this exercise you will do the same for the Gaussian Naive Bayes model. In particular, let X be a d-dimensional continuous random variable and Y be a discrete random variable with range $[K] = \{1, 2, \ldots, K\}$. Moreover, assume $X_j|Y = y \sim \mathcal{N}(\mu_{jy}, \sigma^2_{jy})$ for any $j \in [d]$ and any $y \in [K]$, so that the conditional model $X|Y$ has $2Kd$ parameters. Given data $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ generated by the joint distribution of X and Y, find the maximum likelihood estimators for these parameters (hint: you will need to use *the naive assumption*).

6. In this exercise, you will design a text classification pipeline to categorize movies into four genres: comedy, drama, horror, and action, based on their official textual plot descriptions (`movie-plots-student.csv`). You are free to choose your own strategy, i.e. train/validation splits, tokenization, normalization, vectorization, modeling, validation and other choices are yours (see Lab11 for reference). Describe your methodology in detail, comment on your selected approach, report all intermediate results, and provide your code in a jupyter notebook, ready to be executed (we will run your model on a private hold-out test set). In addition,

please provide a function `test_model(test_data)` that will label our hold-out movie descriptions (given as a list of raw unprocessed strings) according to your processing and modeling pipeline. Top-3 performing submissions (measured by macro-F1 score) will receive 5 extra points towards this homework assignment.

7. Hierarchical clustering:

   (a) Recall the agglomerative clustering algorithm from Lab9: it starts with all samples as their own clusters and proceeds by merging pairs of current clusters based on a specified linkage function; most common linkage functions allow for non-euclidean metrics. Let $\mathcal{D} \subseteq \mathbb{R}^2$ be a dataset of points $A = (0,0), B = (0,1), C = (0,3), D = (2.5,1), E = (3,1), F = (-1,2), G = (4,2)$. Use Manhattan distance together with a complete (maximum) linkage function to produce a full dendrogram describing the agglomerative clustering procedure on $\mathcal{D}$.

   (b) Perform agglomerative clustering with at least 5 different linkage-metric combinations to cluster 5000 MNIST samples (`mnist-sample-X.npy` and `mnist-sample-y.npy` are normalized and flattened, ready-to-use) into 10 groups. For each linkage-metric selection, label the resulting clusters by majority voting and assess the training error. Repeat the procedure for 10-means and 10-means++ methods. Which one worked best?