CAAM/STAT 31310, Autumn 2024, U Chicago

CAAM 31310: Homework 1

Due Oct 15, '24 (11:59 pm ET) on Gradescope

Cite any sources and collaborators; do not copy. See syllabus for policy.

Problem 1

Take X_i to be independent zero-mean random variables with unit variance. Usually, the classical central limit theorem (CLT) is proved by showing that the characteristic function of $Y_n := (1/\sqrt{n}) \sum_{i=1}^n X_i$ converges to the characteristic function of a standard normal. Our task in this problem is to use Stein's identity to prove the CLT. Stein's method has been used to derive computational methods for bounding the distance between two random variables and for a class of sampling algorithms, where the task is to generate samples from a partially specified probability density.

(1) First show that if W is a standard normal RV, for any bounded, differentiable function $f : \mathbb{R} \to \mathbb{R}$, with $\mathbb{E}[Wf(W)], \mathbb{E}f'(W) \leq \infty$, $\mathcal{A}f(W) := f'(W) - Wf(W)$ has zero mean (1 point). This is known as *Stein's lemma*.

(2) For any function $g \in \mathrm{Lip}_1(\mathbb{R})$ (differentiable functions with Lipschitz constant = 1), there is a solution f to *Stein's equation*: $\mathcal{A}f(x) = g(x) - \mathbb{E}g(W)$, where W is a standard normal. (2 points)

(3) Show that a bounded solution f exists that is twice-differentiable with $\|f''\|_\infty \leq 2$ and $\|f'\|_\infty \leq \sqrt{\pi}/2$. (2 points)

(4) Let $\mathbb{E}|X_i|^3 \leq \infty$. Using (2) and (3) above, find a function class $\mathcal{F}$ such that the Wasserstein-1 distance,

$$W^1(Y_n, W) := \sup_{g \in \mathrm{Lip}_1(\mathbb{R})} |\mathbb{E}g(W) - \mathbb{E}g(Y)| \leq \sup_{f \in \mathcal{F}} |\mathbb{E}\mathcal{A}f(Y_n)| \leq C\mathbb{E}|X_i^3|/\sqrt{n}.$$

(2 points)

(5) Use (4) and the converse of Stein's lemma to prove the CLT: $Y_n \xrightarrow{\mathrm{d}} W$. (2 points)

Problem 2

This problem asks you to think about an iterative numerical method as a discrete-time dynamical system (map). Consider the power iteration method for a square, non-singular, diagonalizable matrix $A \in \mathbb{R}^{d \times d}$. For $t \in \mathbb{N}$,

- $v_t \to Av_{t-1}$

(*) $v_{t+1} \to v_{t+1}/\|v_{t+1}\|$.

1. Write down a map $F(x_t) = x_{t+1}$ to describe the above algorithm, where F is defined on a set $M \subseteq \mathbb{R}^d$. (1 point)
2. Is M compact? (1 point)
3. Is F a contraction on M? (1 point)
4. How many fixed points does F have? (1 point) What are they? (1 point)
5. State the assumptions on A so that almost every initial condition converges to a fixed point. (1 point)
6. Under the assumptions in the part above, prove the convergence of almost every iterate to a fixed point of F. (3 points)
7. From here on, consider the power iteration without the normalization step (*). Write the corresponding new map, F, on $\mathbb{R}^d$ (1 point).
8. Give conditions on A for F to be a contraction map (1 point).
9. Give conditions on A for F to be a linear hyperbolic map (1 point).
10. Without the additional conditions in the above two parts (i.e., without hyperbolicity assumptions), describe the asymptotic behavior of all orbits of F. That is, give, with justification, a stable-unstable-center decomposition of $\mathbb{R}^d$ by F. (3 points).

Problem 3

The dataset you are given consists of d-dimensional embeddings of n sentences from (random articles of) simple English Wikipedia. These embeddings, referred to as ET, are calculated with the sentence transformer model from here.

Consider the following alternative way, referred to as $ESVD$, to obtain word embeddings. Construct a matrix, A, of size $n \times n_c$ where n_c is the number of some chosen "context" words found in the data. An entry A_{ij} is set to 1 if the ith sentence contains the jth context word. From the best rank d approximation of A (its SVD), find an embedding for each of the n sentences. You may choose, say the top n_c words (measured by their overall frequency of occurrence) as your context words. Justify any alternative choice for the context words in your answers to the questions below.

(1) For $d = 2$, plot a histogram of your sentence embeddings from ET and $ESVD$ (1 point). Explain your observations, and particularly the dependence of the distribution you see on n_c and on the embedding method (using an SVD or transformers). (3 points)

(2) Check numerically if the classical CLT holds for $d = 2$. Justify your answer with an appropriate convergence plot, e.g., Wasserstein distance between the scaled sample mean and a Gaussian vs no. of samples. (2 points)

(3) As d and n increase, $ESVD$ can produce more identically distributed embeddings. Justify why or not (1 point)

Inputs: The dataset has $n = 10000$ sentences embedded in $d = 64$ dimensions. You can choose a smaller subset of sentences, if you wish, but give your value of n and d in your answer. You have two files: `wiki-embeddings.ob` and `wiki-strings.ob`. You can read `wiki-embeddings.ob` into a list of an 2D array of size $n \times d$ using:

```
import␣pickle
with␣open␣(’wiki-embeddings.ob’,␣’rb’)␣as␣fp:
␣wiki-embeddings␣=␣pickle.load(fp)
```

Similarly, you can read `wiki-strings.ob` into a list of n strings (sentences).

Choosing n_c The sentences are provided so that you can use any means of extracting the n_c context words. For instance, you could use a package to generate word clouds, as long as you cite it and submit your code snippet. The grading will only be based on your explanation of how the choice of context words affects the distribution of the embeddings in 2D, not the choice of context words.