Comparing Decryption of Classical Cipher Text Using Markov Chain Monte Carlo and Principal Component Analysis

Shu-Cheng Zheng

Department of Applied Mathematics, National Taiwan University

Nov, 2024

Outline

Introduction

Introduction - Goal

Table 1 shows output from a typical run of this algorithm (in this case, decrypting the first line of Oliver Twist).

Iteration #	First Line of Decrypted Text
0	LIW PSKMWCL YNLWRDWSY WDKKJ KH KGUXWS LAUEL DQ CIVSGWE FUCJWRE
200	RAS KINJSBR MDRSEHSIM SHNNV NW NGUZSI RPUOR HY BATIGSO LUBVSEO
400	ARS HUNJSPA GDASEBSUG SBNNV NW NMIKSU AFIOA BY PRTUMSO LIPVSEO
600	ARE HLNJEPA KOAESBELK EBNNW NG NMIVEL AFIDA BY PRULMED TIPWESD
800	IME KNSJEPI HUIETBENH EBSSG SW SLOVEN ICODI BY PMANLED ROPGETD
1000	IME GNOJEPI HUIETBENH EBOOK OF OLSVEN ICSDI BY PMANLED RSPKETD
1200	SME GNOJECS HUSETBENH EBOOK OF OLIVEN SPIDS BY CMANLED RICKETD
1400	SME PNOJECS HUSETBENH EBOOK OF OLIVEN SWIDS BY CMANLED RICKETD
1600	SHE MROJECS GUSETBERG EBOOK OF OLIVER SWIDS BY CHARLED NICKETD
1800	SHE PROJECS GUSETBERG EBOOK OF OLIVER SWINS BY CHARLEN DICKETN
2000	SHE PROJECS GUSELBERG EBOOK OF ONIVER SWITS BY CHARNET DICKELT
2200	THE PROJECT GUTENBERG EBOOK OF OLIVER TWIST BY CHARLES DICKENS

Table 1: A sample run of a simple MCMC decryption algorithm.

Introduction - Cryptography

Ciphers can also be categorized in a different way, as classical ciphers and modern ciphers.

- Classical ciphers
  - Substitution
  - Transposition
- Modern ciphers
  - DES (symmetric key)
  - RSA (asymmetric key)

Modern ciphers are correspondingly more complicated and secure than the classical ciphers, and we only consider substitution here.

Introduction - Cryptography (Substitution)

A simple substitution cipher works by replacing each letter with another one. In this paper, we only substitute alphabetic letters; spaces are left untouched and all other non-alphabetic characters are removed.

plain text	THE PROJECT GUTENBERG EBOOK OF OLIVER TWIST
encryption key	XEBPROHYAUFTIDSJLKZMWVNGQC
cipher text	MYR JKSURBM HWMRDERKH RESSF SO STAVRK MNAZM
decryption key	ICZNBKXGMPRQTWFDYEOLJVUAHS
decrypted text	THE PROJECT GUTENBERG EBOOK OF OLIVER TWIST

Table 2: A simple example of a substitution cipher encryption and decryption.

Break Substituted Ciphers by MCMC

Monte Carlo

- Monte Carlo methods are a class of computational techniques used for estimating numerical results through random sampling.
- It is based on the law of large numbers, which states that as the number of samples (random trials) increases, the estimated value converges to the true value.

Example

When faced with the challenge of calculating the integral $\int f(x)dx$ of a complex function $f(x)$, we can employ the Monte Carlo method for an approximation. This approximation involves estimating the integral as follows:

$$\int_a^b f(x)dx \approx \frac{b-a}{N} \sum_{i=1}^{N} f(x_i)$$

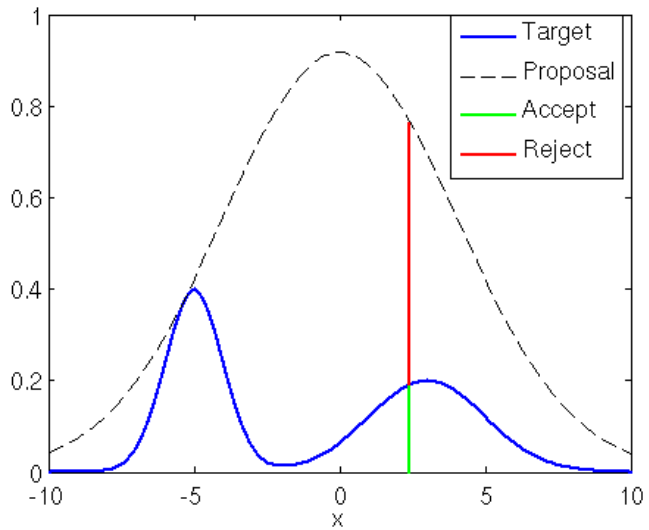
Rejection Sampling

It generates samples from the proposal distribution $q(x)$ and accepts or rejects them based on a comparison with the target distribution $p(x)$.

Step1 Sample $x^{(i)} \sim q(x)$

Step2 Caculate the acceptance ratio is given by $\alpha = \frac{p(x^{(i)})}{Mq(x^{(i)})}$, where M is a constant such that $Mq(x) \geq p(x)$ for all x.

Step3 Given $u \sim U(0,1)$, if $u \leq \alpha$, we accept it; otherwise, we reject it.



Question: Large M values always lead to rejection, while small M values always result in acceptance.

Importance Sampling

- In rejection sampling, when the acceptance rate is too low, a large number of samples are discarded, resulting in too few effective samples. Importance sampling is designed to address the issue of having too few accepted samples in rejection sampling.
- The expected value is estimated as:

$$\begin{aligned}\mathbb{E}_P[g(X)] &= \int g(x)p(x)dx \\ &= \int g(x)\frac{p(x)}{q(x)}q(x)dx \\ &\approx \frac{1}{N}\sum_{i=1}^{N}\frac{p(x_i)}{q(x_i)}g(x_i), \text{ where } x_i \sim q(x) \text{ and } i = 1, \ldots N\end{aligned}$$

- It assigns a weight $\frac{p(x_i)}{q(x_i)}$ to each sample drawn from the proposal distribution and uses these weights to correct the estimates.

Markov chain

A Markov chain is a special type of stochastic process where both time and states are discrete. To satisfy the Markov property, which means that the future is independent of the past given the present, we can express it mathematically as follows:

$$P(x_{t+1} = x|x_1, x_2, \ldots, x_t) = P(x_{t+1}|x_1, x_2, \ldots, x_{t-m}),$$

which represents an m-order Markov property.
The first-order (homogeneous) Markov assumption states that the state of x at time $t+1$ is only dependent on the state of x at time t:

$$P(x_{t+1} = x|x_1, x_2, \ldots, x_t) = P(x_{t+1}|x_t)$$

We can define a transition matrix T as follows to represent it, where $[P_{ij}]$ represents the elements of the matrix:

$$P_{ij} = P(x_{t+1} = j|x_t = i)$$

Stationary Distribution

For a Markov chain, the probability distribution at time $t+1$ can be expressed as:

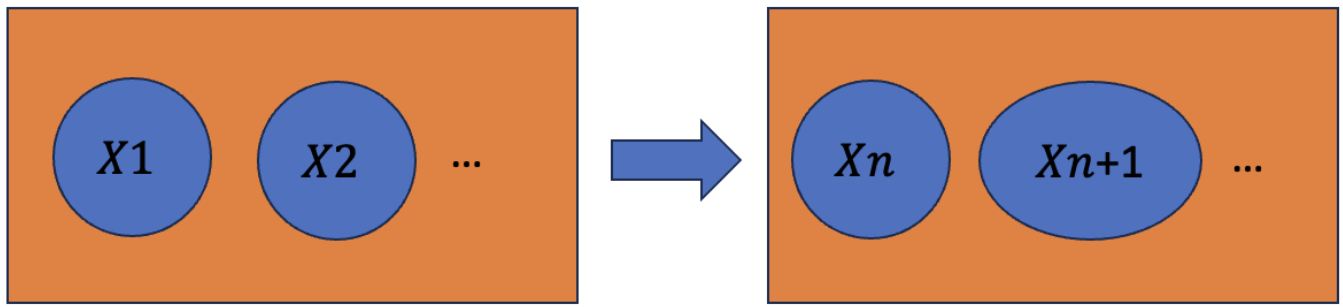
$$\pi_{t+1}(x^*) = \int \pi_t(x) \cdot P(x \mapsto x^*) dx$$

If there exists the above formula that makes the following equation hold:

$$\pi(x^*) = \int \pi(x) \cdot P(x \mapsto x^*) dx$$

Then, we refer to $\{\pi(k)\}_{k=1}^{\infty}$ as the stationary distribution of the Markov chain $\{x_k\}$.
To check if the Markov chain is stationary, we can check whether the **Detailed Balance condition** holds

$$\pi(x) \cdot P(x \mapsto x^*) = \pi(x^*) \cdot P(x^* \mapsto x)$$



Burn in

Sample from p(x)

MCMC - Metropolis-Hastings Sampling

How do we find the transition matrix P_{ij}? We need to find a proposal matrix Q_{ij} such that

$$p(x)q(x^*|x) = p(x^*)q(x|x^*),$$

where $p(x)$ is the target distribution and $q(x^*|x)$ is the proposal distribution for transitioning from state x to state x^*.
To address this, we can introduce a factor on both sides of the equation:

$$p(x)q(x^*|x)\alpha(x, x^*) = p(x^*)q(x|x^*)\alpha(x^*, x),$$

where $\alpha(x, x^*)$ is defined as the acceptance rate, and its value is:

$$\alpha(x, x^*) = \min\left(1, \frac{p(x^*)q(x|x^*)}{p(x)q(x^*|x)}\right)$$

MCMC - Metropolis-Hastings Sampling

Is this definition sufficient to satisfy Detailed Balance?

$$\begin{aligned} p(x)q(x^*|x)\alpha(x,x^*) &= p(x)q(x^*|x)\min\left(1, \frac{p(x^*)q(x|x^*)}{p(x)q(x^*|x)}\right) \\ &= \min\left(p(x)q(x^*|x), p(x^*)q(x|x^*)\right) \\ &= p(x^*)q(x|x^*)\min\left(\frac{p(x)q(x^*|x)}{p(x^*)q(x|x^*)}, 1\right) \\ &= p(x^*)q(x|x^*)\alpha(x^*,x) \end{aligned}$$

Therefore, we have successfully demonstrated that:

$$p(x)q(x^*|x)\alpha(x,x^*) = p(x)q(x^*|x)\alpha(x,x^*)\alpha(x^*,x)$$

Thus, $p(x)$ is a stationary distribution under the transition matrix $q(x^*|x)\alpha(x,x^*)$. It forms a Markov chain, and we can sample our data points from this Markov chain.

Introduction - MCMC

Let $\pi(\cdot)$ be an important possibly-unnormalised density on a state space $\mathcal{X}$. MCMC proceeds by defining an iterative sequence $X_0, X_1, \ldots$ of $\mathcal{X}$-valued variables which converge in distribution to $\pi(\cdot)$. The simplest version of MCMC is the full-dimensional Metropolis algorithm , which proceeds as follows:

- Choose an initial state $X_0 \in \mathcal{X}$.
- For n = 1, 2, 3, ...,
  - Propose a new state $Y_n \in \mathcal{X}$ from some symmetric proposal density $q(X_{n-1}, \ldots)$.
  - Let $U_n \sim$ Uniform[0, 1], independently of $X_0, \ldots, X_{n-1}, Y_n$.
  - If $U_n < (\pi(Y_n)/\pi(X_{n-1}))$, then "accept" the proposal by setting $X_n = Y_n$, otherwise "reject" the proposal by setting $X_n = X_{n-1}$.

Define Score Function

The relevant Markov chain has state space $\mathcal{X}$ consisting of all possible decryption keys. That is, each possible decryption key is a possible state of the Markov Chain. We make use of a long reference text such as a novel. For each pair of characters β_1 and β_2,

- r(β_1, β_2) record the number of times that specific pair appears consecutively in the reference text.
- $f_x(\beta_1, \beta_2)$ record the number of times that pair appears when the cipher text is decrypted using the decryption key x.

To avoid problems from zeroes, we also add one to each of scores.

Example

$\beta_1 = \mathrm{T}$ and $\beta_2 = \mathrm{H}$, r(β_1, β_2) and $f_x(\beta_1, \beta_2)$ record the number of times that specific pair "TH" appears in the reference text and cipher text respectively.

Define Score Function

Score Function

For a particular decryption key x, we then define its score function as

$$\pi(x) = \prod_{\beta_1,\beta_2} r(\beta_1, \beta_2)^{f_x(\beta_1,\beta_2)}.$$

In our computer programs, we compute $\pi(x)$ on a log scale for easy calculation and to avoid numerical errors.

$$\tilde{\pi}(x) = \sum_{\beta_1,\beta_2} f_x(\beta_1, \beta_2) \log r(\beta_1, \beta_2).$$

Intuitively, the score function is higher when the pair frequencies in the decrypted text most closely match those of the reference text, and the decryption key is thus most likely to be correct.

Algorithmn

- Choose an initial state (initial decryption key), and a fixed scaling parameter $m > 0$.
- Repeat the following steps for many iterations (e.g. 20,000 iterations).
  - Given the current state a, propose a new state b from some symmetric density $q(a, b)$.
  - Sample $u \sim Uniform[0, 1]$independently of all other variables.
  - If $u < \left[\pi(b)/\pi(a)\right]^m$ then accept the proposal b by replacing a with b, otherwise reject b by leaving a unchanged.

Break Substituted Ciphers by PCA

Introduction-PCA

Given n data $\mathbf{x_1}, \ldots, \mathbf{x_n} \in \mathbb{R}^p$, Consider the SVD of the matrix

$$(\mathbf{X} - \mathbf{1}\bar{\mathbf{x}}^T)_{(n\times p)} = \mathbf{U}_{(n\times p)}\mathbf{\Lambda}_{(p\times p)}\mathbf{V}^T_{(p\times p)}$$

where $\mathbf{U}^T\mathbf{U} = \mathbf{I}_p$, $\mathbf{V}^T\mathbf{V} = \mathbf{I}_p$, and $\mathbf{\Lambda} = diag(\lambda_1, \ldots, \lambda_p)$ is diagonal matrix of the associated eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_d \geq 0$. Therefore,

$$\mathbf{S} = \frac{1}{n-1}(\mathbf{X} - \mathbf{1}\bar{\mathbf{x}}^T)^T(\mathbf{X} - \mathbf{1}\bar{\mathbf{x}}^T) = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$$

where $\bar{\mathbf{x}} \in \mathbb{R}^p$ is the data mean. Suppose we choose $k(k < p)$ principle components, then the i^{th} principle component score is $y_i = (\mathbf{X} - \mathbf{1}\bar{\mathbf{x}}^T)\mathbf{V_i}$ with the i^{th} principle component $\mathbf{V_i}$.

PCA to the transition matrix

If we view the transition matrix to be our data matrix $\mathbf{X}$ with $p = n$ and we select p principal components, then we can derive that

$$\mathbf{X} - \mathbf{1}\bar{\mathbf{x}}^T = (\mathbf{X} - \mathbf{1}\bar{\mathbf{x}}^T)\mathbf{V}\mathbf{V}^T = \mathbf{Y}\mathbf{V}^T$$

Now we have a reference transition matrix $\mathbf{X_r}$, a plain transition matrix $\mathbf{X_p}$,and a cipher transition matrix $\mathbf{X_c}$

$$\mathbf{Z_r} = (\mathbf{X_r} - \mathbf{1}\bar{\mathbf{x_r}}^T) = \mathbf{Y_r}\mathbf{V_r}^T$$
$$\mathbf{Z_p} = (\mathbf{X_p} - \mathbf{1}\bar{\mathbf{x_p}}^T) = \mathbf{Y_p}\mathbf{V_p}^T$$
$$\mathbf{Z_c} = (\mathbf{X_c} - \mathbf{1}\bar{\mathbf{x_c}}^T) = \mathbf{Y_c}\mathbf{V_c}^T$$

Suppose $\mathbf{Z_r} = \mathbf{Z_p}$

Suppose that the reference transition matrix and the plain transition matrix are exactly the same, denoted as $\mathbf{Z_r} = \mathbf{Z_p}$. In other words, the ciphertexts are generated by encrypting the reference texts through the exchange of several rows and columns. Consequently,

- $\mathbf{Z_r}$ and $\mathbf{Z_c}$ have the same elements but they are out of order.
- The eigenpairs of $\mathbf{S_r}$ and $\mathbf{S_c}$ are same.

To restore the principal components of the ciphertext to match those of the reference, we need to perform a reordering by exchanging the rows and columns. That is

$$\mathbf{Z_r} = \mathbf{E}\mathbf{Z_c}\mathbf{E}^T$$

$$\mathbf{Y_r}\mathbf{V_r}^T = \mathbf{E}\mathbf{Y_c}\mathbf{V_c}^T\mathbf{E}^T$$

Then we have $\begin{cases} \mathbf{Y_r}^T = \mathbf{Y_c}^T\mathbf{E}^T \\ \mathbf{V_r}^T = \mathbf{V_c}^T\mathbf{E}^T \end{cases} \Rightarrow \begin{bmatrix} \mathbf{Y_r}^T \\ \mathbf{V_r}^T \end{bmatrix} = \begin{bmatrix} \mathbf{Y_c}^T \\ \mathbf{V_c}^T \end{bmatrix} \mathrm{E}^T$

Suppose $\mathbf{Z_r} \approx \mathbf{Z_p}$

However, in practice, using the plain text as the reference is unfeasible since it would reveal the solution, defeating the purpose of encryption. Instead, when $\mathbf{Z_r} \approx \mathbf{Z_p}$, we encounter the following situation:

$$\begin{cases} \mathbf{Y_r}^T \approx \mathbf{Y_c}^T \mathbf{E}^T \\ \mathbf{V_r}^T \approx \mathbf{V_c}^T \mathbf{E}^T \end{cases} \Rightarrow \begin{bmatrix} \mathbf{Y_r}^T \\ \mathbf{V_r}^T \end{bmatrix} \approx \begin{bmatrix} \mathbf{Y_c}^T \\ \mathbf{V_c}^T \end{bmatrix} \mathbf{E}^T.$$

Now, we define the three types of distance between $\begin{bmatrix} \mathbf{Y_r}^T \\ \mathbf{V_r}^T \end{bmatrix}$ and $\begin{bmatrix} \mathbf{Y_c}^T \\ \mathbf{V_c}^T \end{bmatrix}$ and our goal is to minimize the distance.

- L_1 norm
- L_2 norm
- Correlation

Algorithmn

- Choose an initial state (initial decryption key), and a fixed scaling parameter $m > 0$.
- Repeat the following steps for many iterations (e.g. 20,000 iterations).
  - Given the current state a, propose a new state b from some symmetric density $q(a, b)$.
  - If $\left[\pi(b)/\pi(a)\right]^m < 1$ then accept the proposal b by replacing a with b, otherwise reject b by leaving a unchanged.

Simulation

Simulation

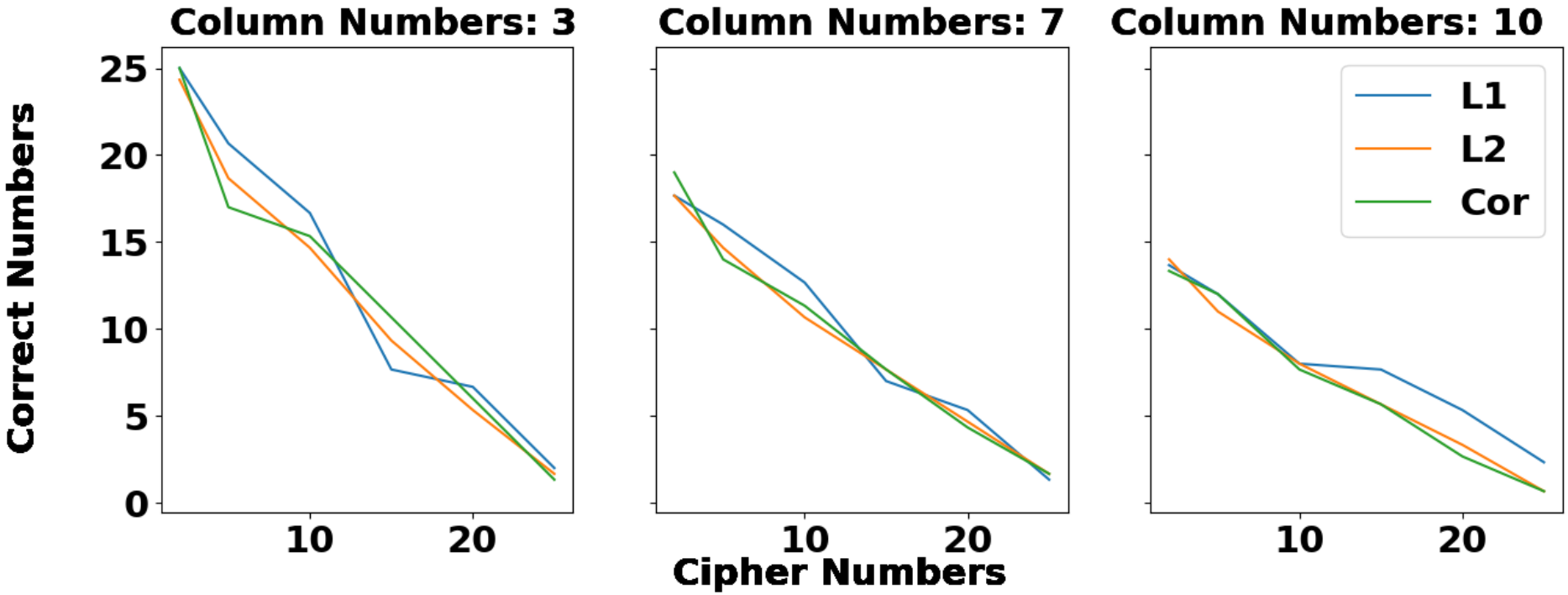


Figure: Decrypted Results of PCA with Different Norms

Simulation

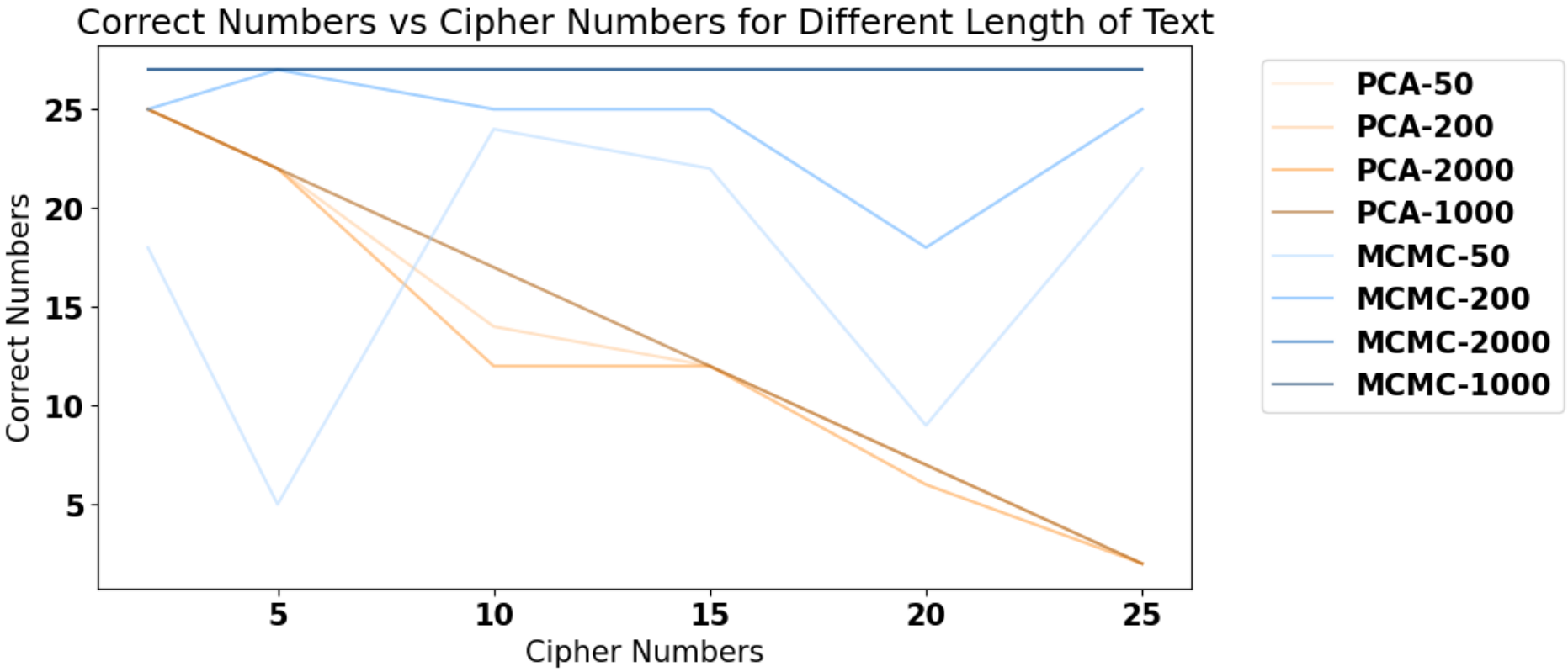


Figure: Decrypted Results of PCA and MCMC with Different Numbers of Texts

Thank You

References I

[1] Jian Chen and Jeffrey S Rosenthal. "Decrypting classical cipher text using Markov chain Monte Carlo". In: *Statistics and Computing* 22.2 (2012), pp. 397–413.