2018年度 修士論文

HTTPS通信における クライアント・サーバ関係のIRM分析

提出日：2019年2月1日

指導：後藤滋樹教授

研究指導名：情報システム工学研究

早稲田大学　基幹理工学研究科　情報理工・情報通信専攻
学籍番号：5117F040-5

佐藤 弘毅

目次

図一覧

表一覧

第 1 章

序論

1.1 研究の背景

HTTP (Hypertext Transfer Protocol) の通信を安全に行うことができる HTTPS のトラヒックが増加している [2]. HTTPS は SSL/TLS を利用して暗号化・認証を行うことで，盗聴や中間者攻撃などを防ぐことができる. HTTPS は当初はインターネットバンキングや電子決済など，セキュリティが重要視されるサービスに採用されていた．しかし，インターネット上のセキュリティやプライバシーの関心の高まりに伴い，一般的な Web サイトにおいても HTTPS を採用する事例が増加している [3]．また，セキュリティ以外の側面でも一般的な Web サイトが HTTPS 化する重要性が増している．例えば，Google は HTTPS に対応しているページを優先的に検索結果に表示することを発表しており [4]，Google Chrome は Chrome 68 から HTTP で Web サイトにアクセスした場合に”Not secure” という警告を表示し [5]，Chrome 70 から HTTP ページでデータ入力した際に赤く警告を表示している [6]．他にも，HTTPS 化のコストは大幅に削減する Let’s Encrypt [7] の登場や，Service Worker [8] のように HTTPS を前提とする技術も登場している．このような背景から，HTTPS トラヒック量は今後も増加すると考えられる．HTTP Archive [9] によると，HTTPS Requests の割合は増加し続けており，2016 年 1 月時点での割合は 24.0 %であったのに対し，2018 年 12 月時点で 79.9 %にまで増加している．

HTTPS は，認証や暗号化を行うことによって安全なデータ転送を提供する．一方で，暗号化された通信はネットワークの品質を管理したり，フィルタリングや異常検出などのトラヒックの正常な監視を困難にする [10]．そのため，HTTPS に関するネットワーク分析技術が求められている．

1.2 研究の目的

本研究では，HTTPS 通信におけるクライアント (ブラウザ等) とサーバ (Web サーバ等) の組に着目する．HTTPS は SSL/TLS を利用して暗号化を行うため，IP ヘッダや TCP ヘッダ，SSL/TLS ハンドシェイク中の情報は暗号化されない．本研究では，SSL/TLS ハンドシェイク中の ClientHello メッセージに含まれる情報を利用する．また，データ解析手法として共クラスタリングの一種である無限関係モデル [11] (IRM, Infinite Relational Model) を用いて，HTTPS 通信におけるクライアントとサーバの関係を分析する．

1.3 論文の構成

本論文は以下の章により構成される．

第 1 章 序論

本論文の概要を述べる．

第 2 章 TLS ハンドシェイク

TLS ハンドシェイクについて説明する．

第 3 章 データ解析手法

本論文で使用するデータ解析手法を説明する．

第 4 章 提案手法

本論文の提案手法を説明する．

第 5 章 実験結果

提案手法に対する実験を行い，その結果を考察する．

第 6 章 結論

本論文の結論を述べるとともに, 残された課題を示す．

第 2 章

TLSハンドシェイク

2.1　TLSハンドシェイクの概要

TLS プロトコルは，主に TLS ハンドシェイクプロトコルと TLS レコードプロトコルの 2 つから成る．TLS ハンドシェイクプロトコルは，サーバとクライアントが相互に認証し，暗号化に用いる暗号化アルゴリズムと暗号鍵の共有を行い，セッションを確立するために用いられる．TLS レコードプロトコルは，確立されたセッションにおいて共通鍵を用いてデータを暗号化し，安全なデータ転送を提供する．2019 年 1 月時点での最新版は TLS1.3 [12] であるが，最も広く普及しているバージョンである TLS1.2 [13] における TLS ハンドシェイクのメッセージフローを図 2.1 に示す．

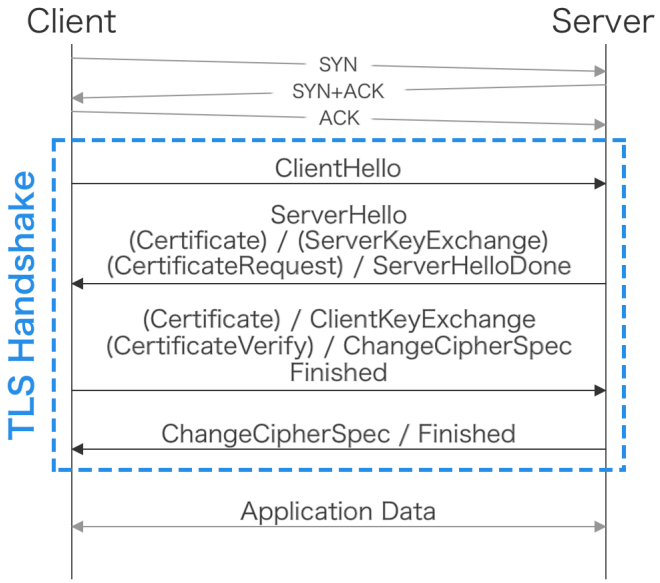


図 2.1: TLS1.2 におけるフルハンドシェイクのメッセージフロー

2.2 Server Name Indication

SNI (Server Name Indication) は，RFC3546 [14] で定義された TLS の拡張機能である．TLS では，ハンドシェイクにおいてクライアントはサーバから証明書を受け取り，証明書の改ざんの有無や，証明書に書かれているホスト名とアクセスしようとしているホスト名が一致するかを確認する．一方で，単一の IP アドレスで複数のドメイン名を運用する名前ベースバーチャルホストを利用している場合には，TLS ハンドシェイクを行う段階でサーバはクライアントがどのホストにアクセスするか判断できない [15]．そのため，名前ベースバーチャルホストではホスト名ごとに異なる証明書を使い分けることができない．SNI はこの課題を解決するための機能である．SNI は TLS ハンドシェイクの ClientHello メッセージにおいて，クライアントがサーバにアクセスしたいホスト名を含むことができる．これにより，サーバはクライアントがどのホストにアクセスするか判断して，1 つの IP アドレスで複数の SSL 証明書を使い分けることが可能となる．

SNI は暗号通信を始める前の ClientHello メッセージに含まれるため，平文でホスト名を送ることになる．したがって，SNI の値はネットワーク上で観測することが可能である [16]．本研究では，SNI の値を用いてクライアントがどのホストにアクセスしているかを判断する．

2.3 Cipher Suite

暗号スイート (Cipher Suite) は TLS で用いるアルゴリズムのセットである．鍵交換アルゴリズムと鍵認証方式，暗号化に用いる共通鍵暗号およびその鍵長と暗号モード，メッセージ認証符号が含まれる．例えば TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 = { 0x00,0x67 }; の場合，鍵交換アルゴリスムに Ephemeral Diffie-Hellman，署名アルゴリズムに RSA，暗号化に用いる共通鍵暗号に 128bit 鍵長で暗号モードが CBC の AES，メッセージ認証符号に SHA-256 を用いる．TLS ハンドシェイクでは，はじめに ClientHello メッセージでクライアントが対応している暗号スイートのリストを送信する．サーバは送信された暗号スイートのリストの中から 1 つを選び，ServerHello メッセージで選択した暗号スイートを送信する．

SNI と同様に，ClientHello メッセージに含まれる暗号スイートのリストはネットワーク上で観測可能である．ブラウザの種類やバージョンによって対応している暗号スイートは異なるため，暗号スイートのリストはクライアントの識別に用いられる [17]．なお，一般的に HTTP 通信におけるクライアントの識別には，HTTP リクエストに含まれる User-Agent ヘッダが使われるが，HTTPS 通信では HTTP リクエストは暗号化されるため User-Agent を観測できない．本研究においても，暗号スイートのリストをクライアントの識別に使用する．

第 3 章

データ解析手法

本章では，本研究で用いるデータ解析手法について説明する．

3.1 クラスタリング

クラスタリングとは，データ解析で使用される手法の一つで，入力データ間の類似度 (または非類似度) に基づいて，データを複数のクラスタ (グループ) に分類するものである．教師データを必要としないデータ分類手法である．クラスタリングには様々な手法が提案されているが，特定のクラスタ数に分類する非階層的な手法と，類似度の高い順にクラスタを結合していく階層的な手法に大別することができる．前者は非階層型クラスタリングと呼ばれ，代表例として k-means 法，k-means++法，x-means 法がある．後者は階層型クラスタリングと呼ばれ，代表例として単連結法，完全連結法，ウォード法がある．本研究では，非階層型クラスタリングの一つである x-means 法をデータ解析手法の一つとして用いる．そこで本節では，k-means 法，k-means++法，x-means 法について説明する．

3.1.1 k-means 法

k-means 法は，d 次元の N 個のデータ $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2 \ldots, \boldsymbol{x}_N\}$ を，データ間の類似度 (距離) を尺度に，あらかじめ定めた K 個のクラスタ $C = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_K\}$ に分類する手法である．各クラスタの中心点 (セントロイド) と，クラスタ内の各データの間の 2 乗ユークリッド距離の総和が最小になるように最適化を行う．k 番目のクラスタ C_k の中心点を $\boldsymbol{\mu}_k$ とすると，評価関数 $J(C)$ は以下のように表すことができる [18]．

$$J(C) = \sum_{k=1}^{K} \sum_{\boldsymbol{x}_i \in \boldsymbol{c}_k} \|\boldsymbol{x}_i - \boldsymbol{\mu}_k\|^2 \tag{3.1}$$

k-means 法は，式 (3.1) が最小になるように最適化する．k-means 法のアルゴリズムを以下に示す．

1. **初期化:** N 個のデータをランダムに K 個のクラスタに振り分け，各クラスタの平均ベクトル (重心) を求め，$\boldsymbol{\mu}_k$ $(k = 1, \ldots, K)$ とする．
2. **クラスタの再編成:** 各データを最も距離が近い中心点を持つクラスタに割り当てる．
3. **重心の更新:** 各クラスタの重心 $\boldsymbol{\mu}_k$ を更新する．
4. **繰り返し:** 上記の 2. と 3. を収束する (状態変化がなくなる) まで繰り返す．

k-means 法の最適解は NP 困難であり，アルゴリズムの収束先は初期値に依存する．したがって，最適解に近い解を得るためには，何回か初期値を変えて実行する必要がある．

3.1.2　k-means++法

k-means++法は，k-means 法の初期値の選択を改良したアルゴリズムである [19]．初期値の選択を改良することにより，近似解の精度の向上と収束スピードの向上が期待される．なお，k-means 法と同様に，クラスタ数をあらかじめ指定する必要がある．k-means++法の初期値の選択アルゴリズムは以下である．

1. データの中からランダムに 1 つ選択し，あるクラスタの重心とする．
2. 新しいクラスタの重心を，以下の確率分布に従って選択する．
   ただし $D(\boldsymbol{x}_i)$ はデータ $\boldsymbol{x}_i$ とすでに選択された重心の中で最も距離が近い重心との距離を表す．
$$\frac{D(\boldsymbol{x}_i)^2}{\sum_{\boldsymbol{x}_i \in X} D(\boldsymbol{x}_i)^2} \tag{3.2}$$
3. 指定したクラスタ数 K 個の重心が選択されるまで 2. を繰り返す．

3.1.3　x-means 法

x-means 法は，k-means 法の拡張アルゴリズムである [20, 21]．k-means 法や k-means++法はクラスタ数をあらかじめ指定する必要があるのに対し，x-means 法は最適なクラスタ数を自動推定することが可能である．x-means 法は，k-means 法の逐次繰り返しと BIC (Bayesian Information Criterion) に基づく分割停止により，クラスタ数を自動で推定する．x-means 法のアルゴリズムの概要を以下に示す．

1. クラスタ数の初期値 k_0 (特に指定しなければ 2) を定め，$k = k_0$ として k-means 法を適用し，分割後のクラスタを $C_1, C_2, \ldots, C_{k_0}$ とする．
2. クラスタ C_i に対して $k = 2$ として k-means 法を適用し，分割後のクラスタを C_i^1, C_i^2 とする．
3. 分割前の BIC と分割後の BIC' を計算し，BIC $\geq$ BIC' であれば分割しない．
4. BIC $<$ BIC' であれば C_i^1, C_i^2 に対して 2. から 4. を行う．
5. すべてのクラスタの分割が終了するまで 2. から 4. を繰り返す．

3.2　無限関係モデル

顧客と商品のように，異なる種類のオブジェクトを同時にクラスタリングする手法を共クラスタリングという．無限関係モデルは，Kemp らにより提案された共クラスタリングを実現する手法である [11]．無限関係モデルはクラスタ数を事前に決定する必要がなく，ノンパラメトリックベイスモデルにより最適なクラスタ数を自動で推定することが可能である．無限関係モデルによるクラスタリングの例を図 3.1 に示す．

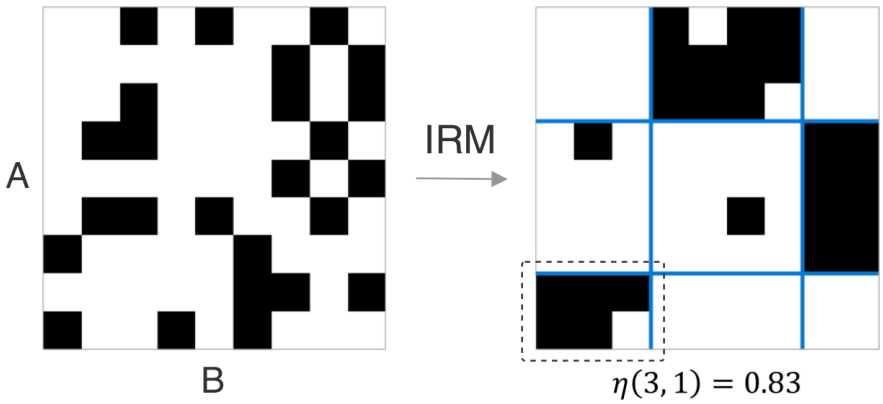


図 3.1: 無限関係モデルによるクラスタリングの例

無限関係モデルは，オブジェクト間のつながりを表す関係データを解析するのに用いられる．オブジェクトの集合 $A = \{1, 2, \ldots, K\}$ とオブジェクトの集合 $B = \{1, 2, \ldots, L\}$ の間の関係データ R は，オブジェクト i, j 間に関係が存在する場合は $R_{ij} = 1$ (黒)，しない場合は $R_{ij} = 0$ (白) として 2 値行列で表現できる．無限関係モデルは，関係データ R の行と列を並び替えることでクラスタリングを行う．A の各オブジェクトの所属クラスタを表す潜在変数を $\boldsymbol{s}^1 = \{s_1^1, s_2^1, \ldots, s_K^1\}$，

B の各オブジェクトの所属クラスタを表す潜在変数を $\boldsymbol{s}^2 = \{s_1^2, s_2^2, \ldots, s_L^2\}$ とする．無限関係モデルは，式 (3.3) で表される $(\boldsymbol{s}^1, \boldsymbol{s}^2)$ の事後確率 P を最大化する $(\boldsymbol{s}^1, \boldsymbol{s}^2)$ を求める問題に帰着する．

$$P(\boldsymbol{s}^1, \boldsymbol{s}^2|R) = \frac{P(R|\boldsymbol{s}^1, \boldsymbol{s}^2)P(\boldsymbol{s}^1)P(\boldsymbol{s}^2)}{P(R)} \tag{3.3}$$

無限関係モデルのアルゴリズムの概要を以下に示す [22].

1. **初期化:** 行・列の各々のオブジェクトにクラスタを割り当て $\boldsymbol{s}^1, \boldsymbol{s}^2$ を初期化する．
   初期化時点での行クラスタ数を c_1，列クラスタ数を c_2 とする．
2. **所属クラスタの更新:** 行オブジェクト $i = 1, 2, \ldots, K$ に対して以下を行う．
   - 所属クラスタ s_i^1 を更新するため，現在の所属クラスタから除外する．
   - これにより空クラスタが発生した場合は $c_1 \leftarrow (c_1 - 1)$ とする．
   - $s_i^1 = k(k = 1, \ldots, c_1 + 1)$ に対して事後確率を計算し，s_i^1 の値を決定する．
   - $s_i^1 = c_1 + 1$ となった場合は $c_1 \leftarrow (c_1 + 1)$ とする．

   同様の処理を列オブジェクト $i = 1, 2, \ldots, L$ に対しても行う．
3. **事後確率最大化:** 現時点での $\boldsymbol{s}^1, \boldsymbol{s}^2$ の値を用いて，式 (3.3) により事後確率 P を計算する．
   $P > P_{max}$ であれば $P_{max}, \boldsymbol{s}^1, \boldsymbol{s}^2$ の値を更新する．
4. **繰り返し:** 上記の 2. と 3. を収束する (P_{max} が更新されなくなる) まで繰り返す．

無限関係モデルにおいて，A の k 番目のクラスタと B の l 番目のクラスタが成す矩形領域において R_{ij} が 1 となる確率を η_{kl} と表す．この値はクラスタ k とクラスタ l の関係の強さを表す．矩形領域内の $R_{ij} = 1, R_{ij} = 0$ の数をそれぞれ $m_{kl}, \overline{m}_{kl}$ とすると，η_{kl} は無限関係モデルのハイパーパラメータ β を用いて式 (3.4) で表される．

$$\eta_{kl} = \frac{m_{kl} + \beta}{m_{kl} + \overline{m}_{kl} + 2\beta} \tag{3.4}$$

第 4 章

提案手法

4.1 提案手法の概要

本稿では，無限関係モデルを利用して HTTPS 通信を分類する手法を提案する．本研究は，HTTPS 通信におけるクライアント (ブラウザ等) とサーバ (Web サーバ等) の組を対象とする．提案手法の概要を図 4.1 に示す．提案手法は，関係データの作成，x-means 法による期間の分類，無限関係モデルの適用の 3 つのステップから構成される．以下に各ステップの詳細を順番に説明する．

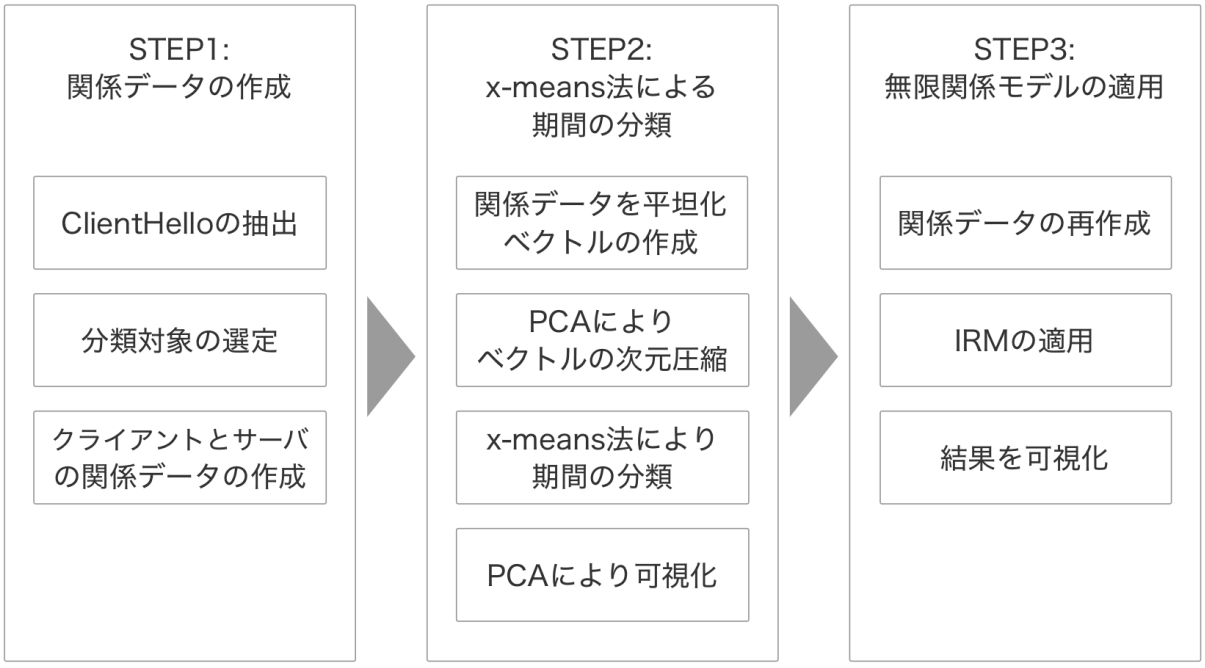


図 4.1: 提案手法の概要

4.2 STEP1: 関係データの作成

本手法では，HTTPS 通信の際に行われる TLS ハンドシェイクに着目し，その中の ClientHello メッセージを利用する．はじめに，HTTPS 通信の中から ClientHello メッセージを抽出し，ClientHello メッセージの中に含まれる送信元 IP アドレスと SNI (Server Name Indication) の値を抽出する．ここで，抽出した送信元 IP アドレスをクライアント，SNI の値をホストとする．次に，分類対象とするクライアントの集合とサーバの集合を選定する．これは関係データがスパースな行列になるのを防ぐためである．スパースなデータ行列に対して無限関係モデルを適用すると，多数の小さなクラスタが生成され，重要なクラスタを見つける際の障害になる [23]．本研究では，サーバごとのアクセス回数の順位を算出して上位 500 個を分類対象のサーバとし，対象のサーバに対して 1 回以上のアクセスをした 328 個の IP アドレスを分類対象のクライアントとした．続いて，期間ごとにクライアントとサーバの関係データを作成する．期間中に分析対象となる各クライアントが分析対象となる各サーバに対して 1 回以上アクセスした場合は 1，そうでない場合は 0 とする．本研究では 3 時間帯 $\times$ 7 日 $=$ 21 個の期間を設定したため，21 個の関係データが作成される．

4.3 STEP2: x-means 法による期間の分類

STEP2 では，STEP1 で作成した期間ごとの関係データを x-means 法によりクラスタリングする．はじめに，2 次元の関係データを 1 次元に平坦化して期間ごとのベクトルを作成する．図 4.2 に例を示す．次に，期間ごとに作成したベクトルに対して主成分分析 (PCA, Principal Component Analysis) を行い，ベクトルの次元を削減する．図 4.3 に例を示す．本研究では，クライアントの数が 328 個，ホストの数が 500 個のため，関係データをベクトルにした際のベクトルの次元数は $328 \times 500 = 164,000$ となるが，これを主成分分析により 21 次元まで削減した．これは，期間の数が 21 個であることに起因する．期間の数を N 個とすると，主成分分析の第 N 主成分までの累積寄与率は 1.0 になるため，N 次元に圧縮しても情報の損失はないと考えられる．

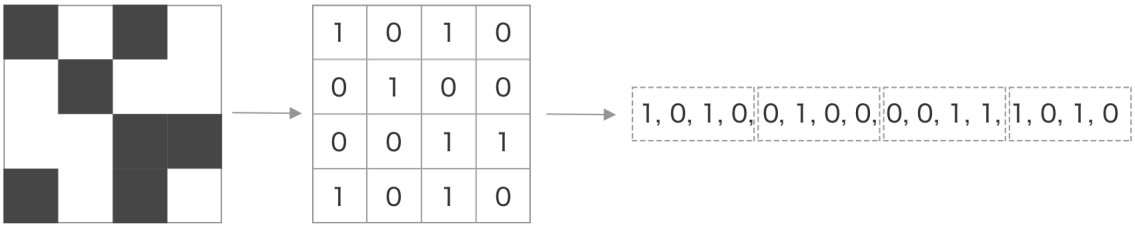


図 4.2: 期間ごとのベクトル作成の例

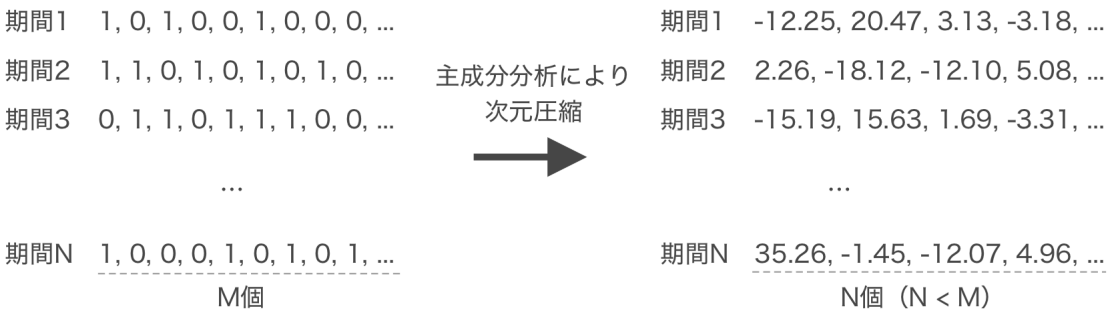


図 4.3: 主成分分析によるベクトルの次元圧縮の例

次に，次元圧縮したベクトルを x-means 法により分類する．これにより，関係データが曜日や時間帯などによって特徴が異なるのかを調査する．最後に，ベクトルを主成分分析により 2 次元に圧縮し，クラスタリングの結果を可視化する．

4.4 STEP3: 無限関係モデルの適用

STEP3 では，STEP2 で得られたクラスタごとに関係データを作成し直し，無限関係モデルを適用する．STEP2 において x-means 法により分類された期間のクラスタを期間グループとここで呼ぶことにする．はじめに，期間グループごとに関係データを作成し直す．つまり，期間グループ中の期間において，分析対象となる各クライアントが分析対象となる各サーバに対して 1 回以上アクセスした場合は 1，そうでない場合は 0 とする．次に，期間グループごとに作成した関係データに対して無限関係モデルを適用し，共クラスタリングを行う．最後に，3.2 節で説明した無限関係モデルにおけるクラスタ間の関係の強さを表すパラメータである η を可視化する．

第 5 章

実験結果

本章では実際のデータを対象に，4 章で説明した提案手法を適用して結果を示し考察する．はじめに，実験で用いたデータについて 5.1 節で説明する．続いて，5.2 節で STEP2 (x-means 法による期間の分類) の結果を示し，5.3 節で STEP3 (無限関係モデルの適用) の結果を示す．

5.1 実験に用いたデータ

実験に使用するデータとして，実ネットワーク上を流れるパケットを収集した．測定期間は 2018 年 7 月 2 日から 2018 年 7 月 8 日までの毎日 09:00–10:00，13:00–14:00，19:00–20:00 の時間帯である．各期間において観測された ClientHello メッセージの数を表 5.1 に示す．

表 5.1: 各期間における ClientHello メッセージ数

日付＼時間帯	09:00–10:00	13:00–14:00	19:00–20:00
2018/07/02 (月)	125,408	90,839	97,051
2018/07/03 (火)	136,343	82,365	95,196
2018/07/04 (水)	113,251	81,308	92,877
2018/07/05 (木)	138,576	84,390	110,790
2018/07/06 (金)	144,435	105,669	115,080
2018/07/07 (土)	115,080	111,698	86,647
2018/07/08 (日)	94,855	96,871	58,503

4.2 節で述べたように，サーバごとのアクセス回数の順位を算出して上位 500 個を分類対象のサーバとし，対象のサーバに対して 1 回以上のアクセスした 328 個のクライアントを分類対象のクライアントとした．サーバごとのアクセス回数の順位，アクセス回数，およびその累積相対度数の主要な値を表 5.2 に示す．分類対象の 500 個のサーバへの通信により，パケット全体の 7 割以上を占めていることがわかる．

表 5.2: サーバごとのアクセス回数の順位と累積相対度数

アクセス順位	アクセス回数	累積相対度数
1	84,554	0.039
4	31,205	0.103
15	13,406	0.203
39	6,135	0.300
87	3,631	0.401
155	2,576	0.500
261	1,635	0.600
447	844	0.700
855	334	0.800
2248	77	0.900
32,755	1	1.000

5.2 x-means 法による期間の分類結果

4.3 節で示した手法により，期間ごとに作成した関係データを元としたベクトルを使用して，x-means 法によりクラスタリングを行った．クラスタリングの結果を表 5.3 に示す．また，主成分分析を用いてベクトルを 2 次元に削減して平面上に図示した結果を図 5.1 に示す．横軸は第一主成分，縦軸は第二主成分である．朝，昼，夜の時間帯や，平日・休日の曜日によってクラスタが分類された．このことから，クライアントとサーバの関係データは時間帯や平日と休日によって特徴があることがわかる．従来のトラヒック分析においても，トラヒック量の傾向が一日のうちの時間帯により変化することや，平日と土日で変化することは知られていた．表 5.3 の結果はそれらに合致する．

表 5.3: x-means 法によるクラスタリング結果

日付＼時間帯	09:00–10:00	13:00–14:00	19:00–20:00
2018/07/02 (月)	クラスタ 1	クラスタ 2	クラスタ 3
2018/07/03 (火)	クラスタ 1	クラスタ 2	クラスタ 3
2018/07/04 (水)	クラスタ 1	クラスタ 2	クラスタ 3
2018/07/05 (木)	クラスタ 1	クラスタ 2	クラスタ 3
2018/07/06 (金)	クラスタ 1	クラスタ 2	クラスタ 3
2018/07/07 (土)	クラスタ 4	クラスタ 4	クラスタ 5
2018/07/08 (日)	クラスタ 5	クラスタ 5	クラスタ 5

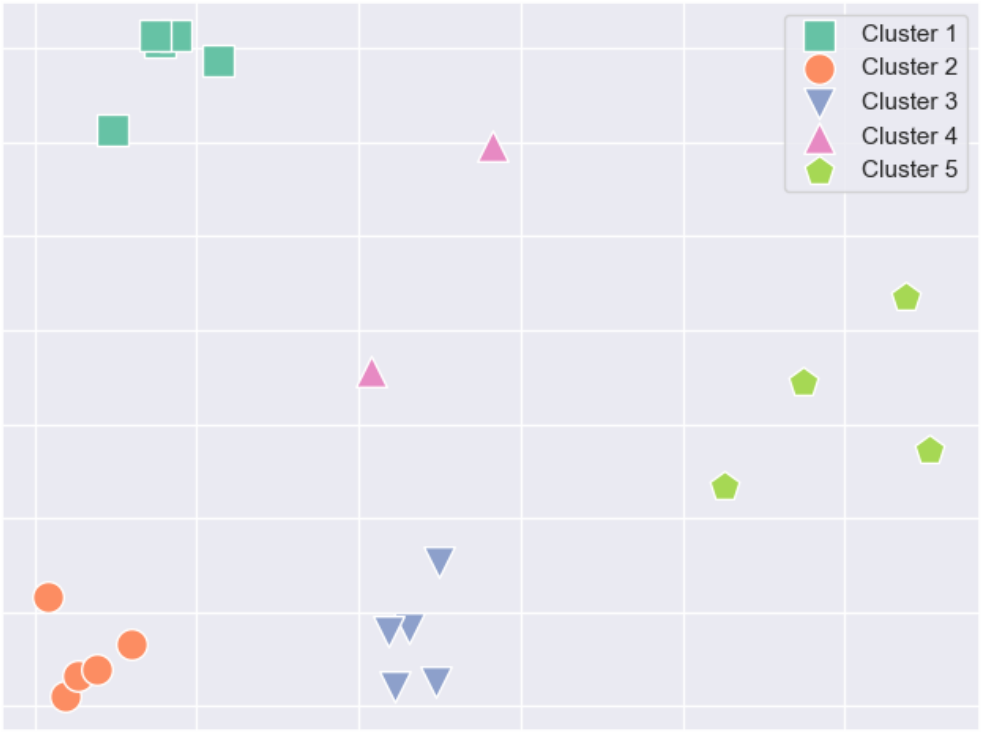
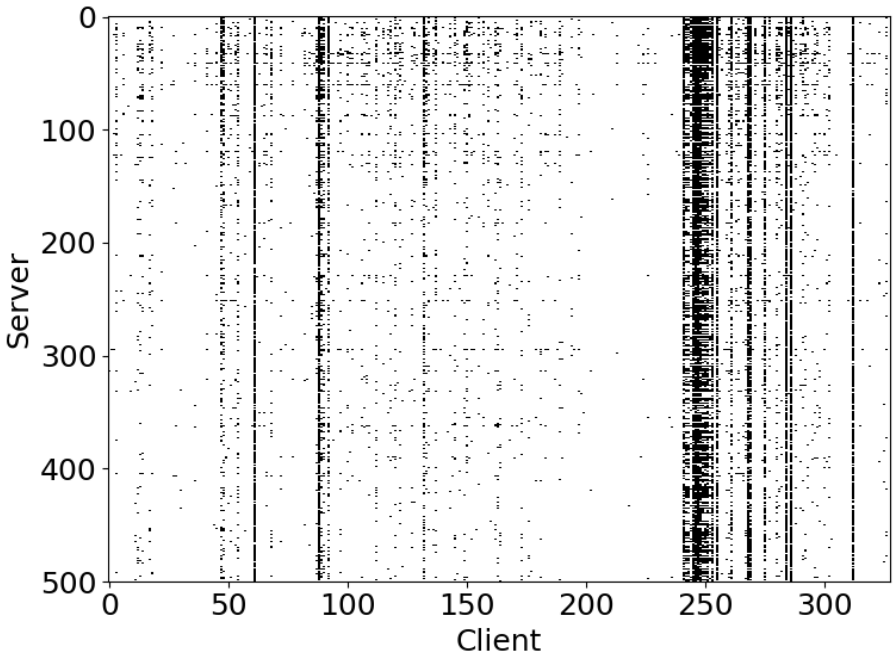


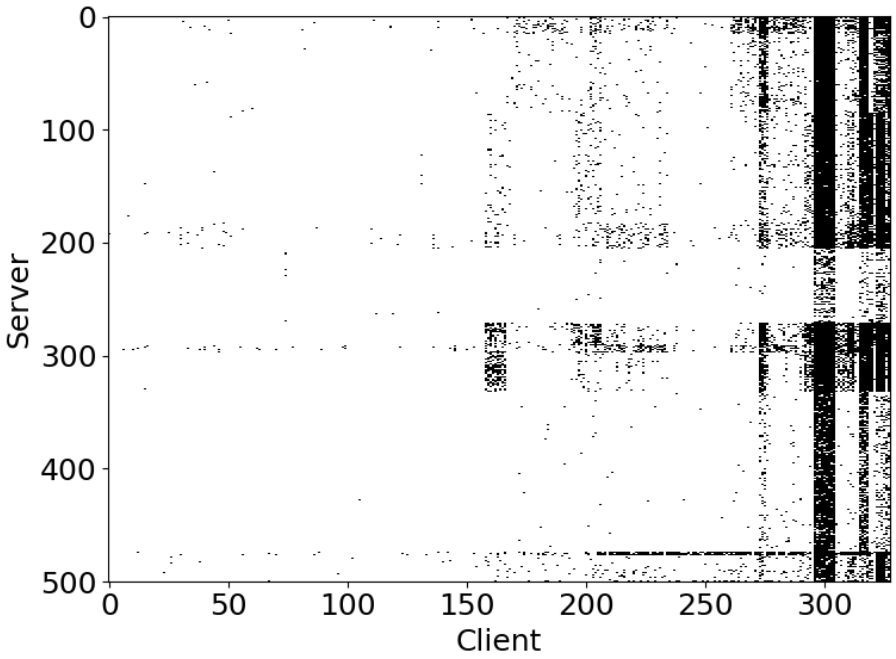
図 5.1: x-means 法によるクラスタリング結果

5.3　無限関係モデルの適用結果

4.4 節で説明した手法により，5.2 節で示したクラスタ (期間グループ) ごとに関係データを作り直し，無限関係モデルを適用した．無限関係モデルを適用する前後の関係データを図 5.2 から図 5.6 に示す．縦軸はサーバの，横軸はクライアントの番号である．また，無限関係モデルによりクライアントとサーバともに並び替えられるため，適用前と適用後で番号は対応していないことに注意が必要である．期間グループによってグラフに特徴があることがわかる．また，STEP2 では関係データを平坦化して主成分分析により次元圧縮してから計算しているため，クライアントとサーバの関係を保持できないのに対し，STEP3 では関係データがクライアントとサーバの関係を保っていることがわかる．

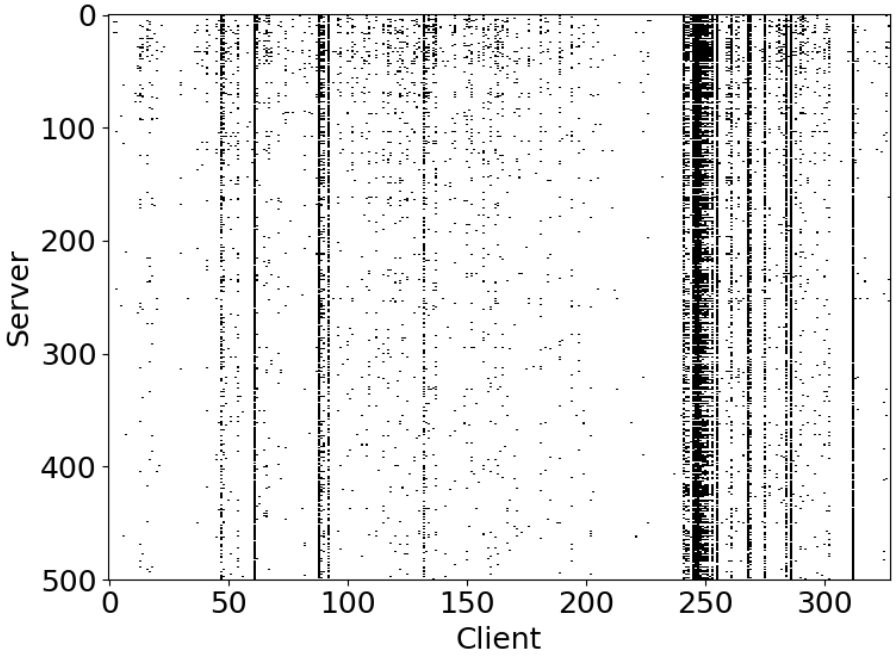


(a) 無限関係モデル適用前の関係データ

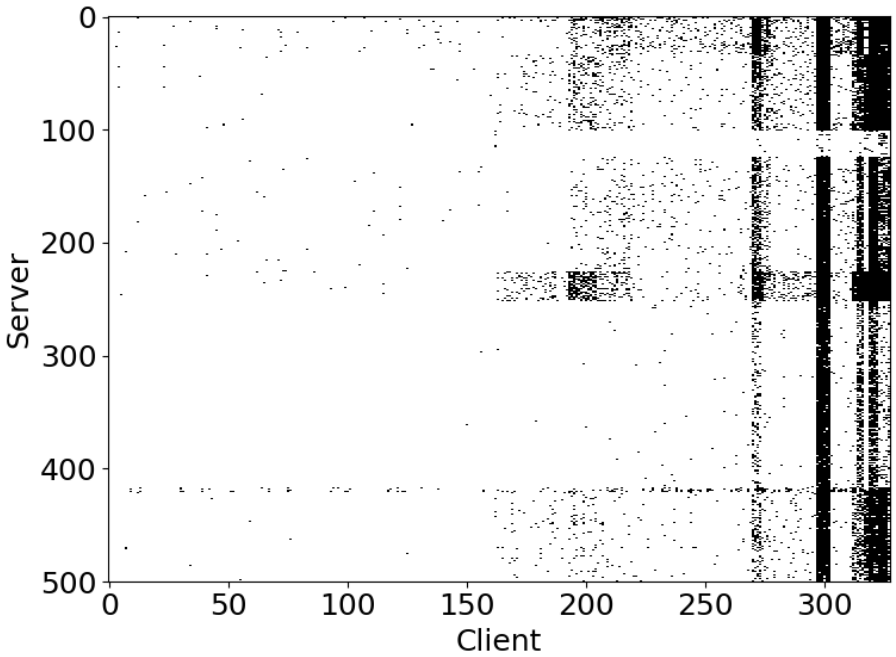


(b) 無限関係モデル適用後の関係データ

図 5.2: クライアントとサーバの関係データ (期間グループ 1)

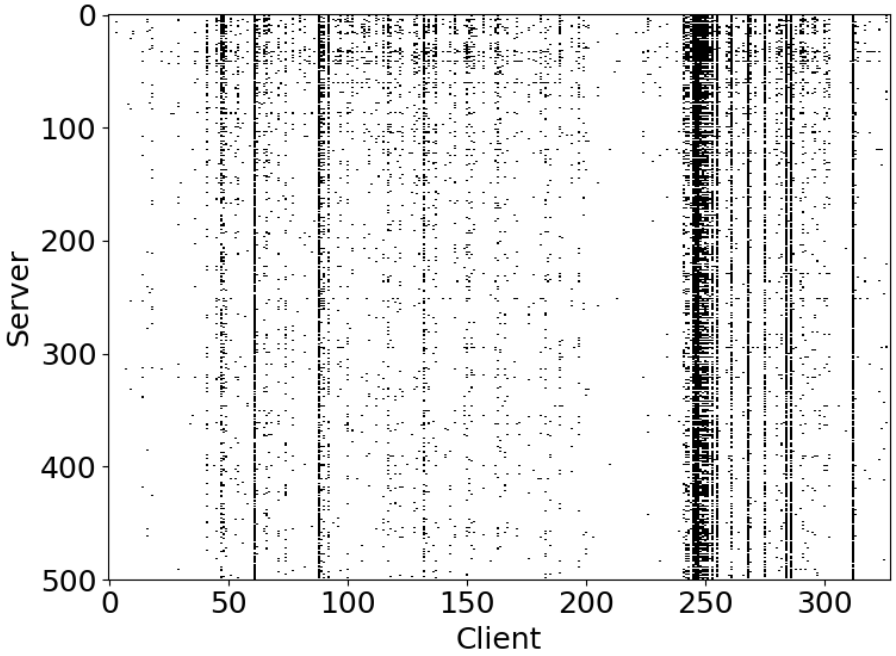


(a) 無限関係モデル適用前の関係データ

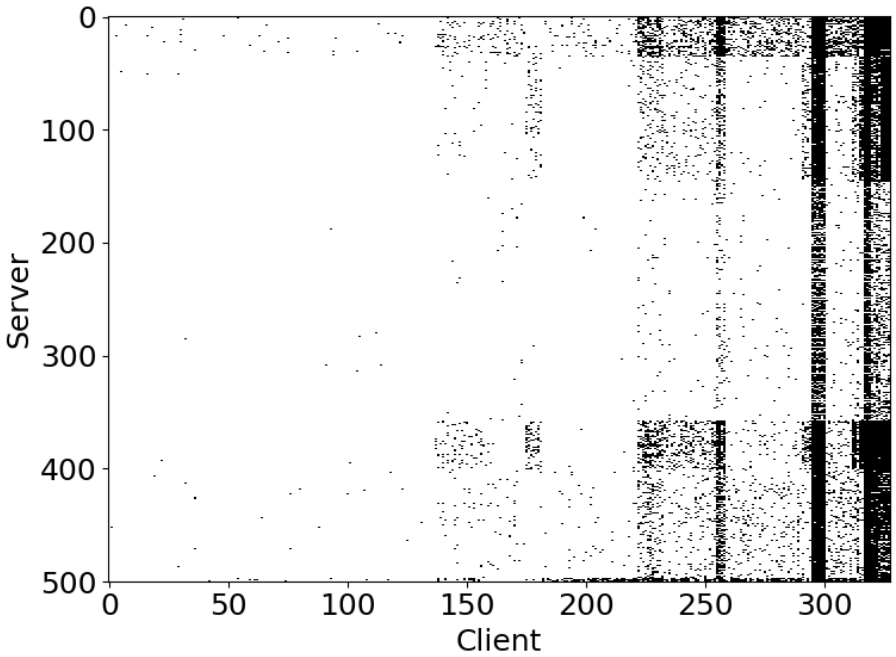


(b) 無限関係モデル適用後の関係データ

図 5.3: クライアントとサーバの関係データ (期間グループ 2)

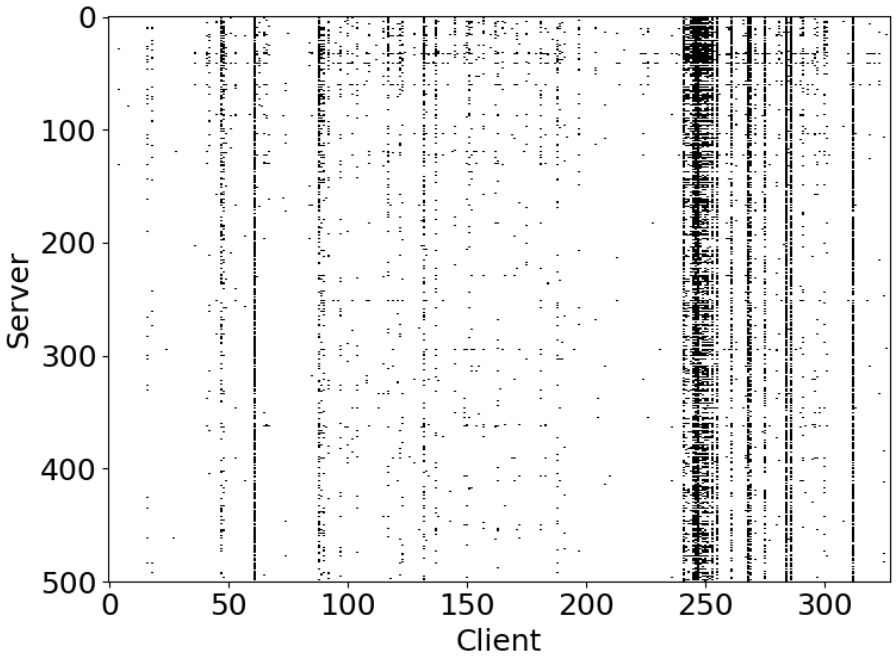


(a) 無限関係モデル適用前の関係データ

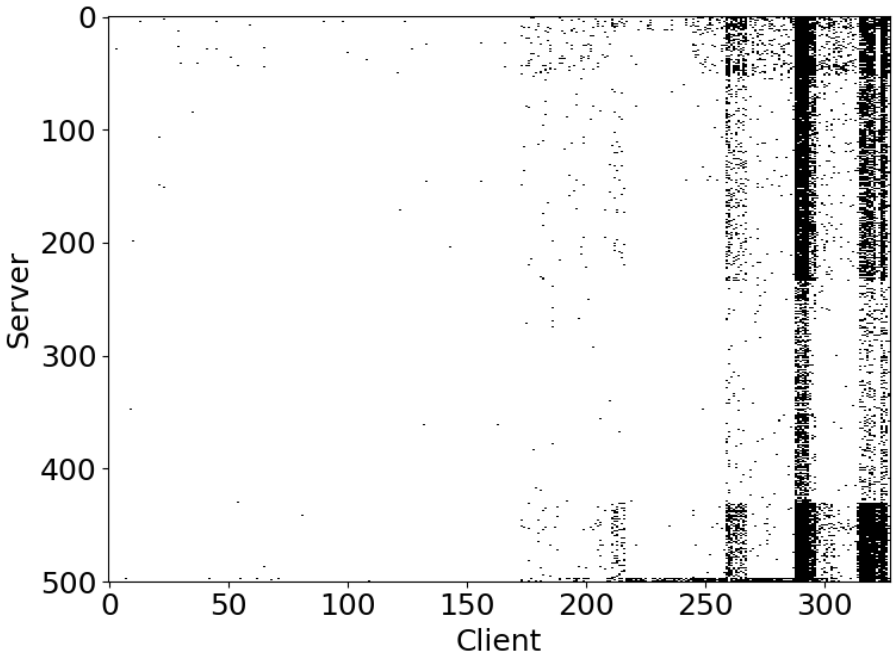


(b) 無限関係モデル適用後の関係データ

図 5.4: クライアントとサーバの関係データ (期間グループ 3)

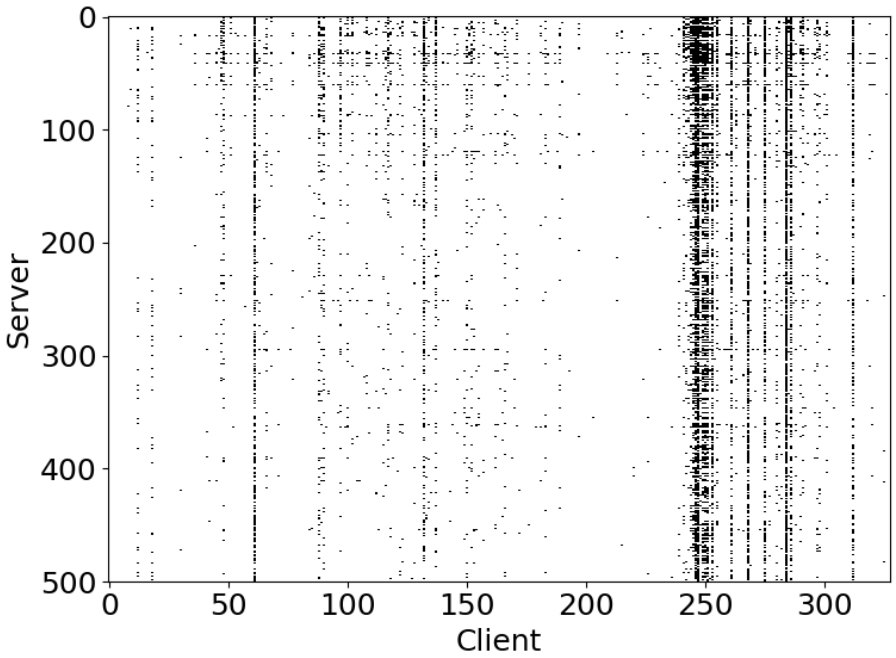


(a) 無限関係モデル適用前の関係データ

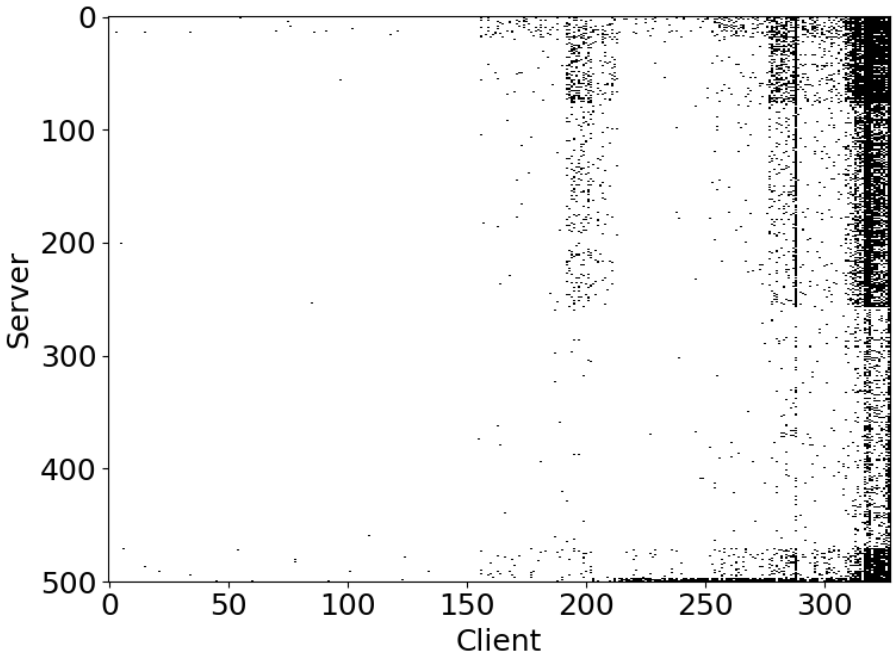


(b) 無限関係モデル適用後の関係データ

図 5.5: クライアントとサーバの関係データ (期間グループ 4)



(a) 無限関係モデル適用前の関係データ



(b) 無限関係モデル適用後の関係データ

図 5.6: クライアントとサーバの関係データ (期間グループ 5)

次に，無限関係モデルにより共クラスタリングした結果を示す．無限関係モデルにより生成されたクラスタの数を表 5.4 に示す．期間グループ 1 は他の期間グループと比較して，生成されたクラスタ数がサーバ・クライアントともに多いことがわかる．また，各クラスタに分類されたサーバの数を表 5.5 に，クライアントの数を表 5.6 にそれぞれ示す．なお，表 5.5，表 5.6 において “–” は該当するクラスが存在しないことを表す．どの期間においても，サーバ・クライアントともに，所属する要素の数が多い大きなクラスタと，所属する要素の数が少ない小さなクラスタに分かれた．

表 5.4: 無限関係モデルにより生成されたクラスタの数

期間グループ	サーバ	クライアント
期間グループ 1	12	22
期間グループ 2	9	16
期間グループ 3	6	19
期間グループ 4	6	18
期間グループ 5	6	16

表 5.5: 無限関係モデルにより各クラスタに分類されたサーバの数

クラスタ	期間グループ				
	グループ 1	グループ 2	グループ 3	グループ 4	グループ 5
クラスタ 1	16	4	36	13	19
クラスタ 2	69	30	111	40	58
クラスタ 3	97	67	211	181	180
クラスタ 4	24	23	43	196	213
クラスタ 5	65	102	96	67	27
クラスタ 6	20	26	3	3	3
クラスタ 7	6	165	–	–	–
クラスタ 8	35	4	–	–	–
クラスタ 9	142	79	–	–	–
クラスタ 10	3	–	–	–	–
クラスタ 11	22	–	–	–	–
クラスタ 12	1	–	–	–	–

表 5.6: 無限関係モデルにより各クラスタに分類されたクライアントの数

クラスタ	期間グループ				
	グループ 1	グループ 2	グループ 3	グループ 4	グループ 5
クラスタ 1	158	162	137	173	156
クラスタ 2	9	31	23	38	36
クラスタ 3	27	12	15	6	11
クラスタ 4	8	15	7	28	11
クラスタ 5	5	50	40	14	40
クラスタ 6	28	4	11	2	23
クラスタ 7	26	4	22	7	11
クラスタ 8	12	19	2	20	1
クラスタ 9	3	6	2	6	20
クラスタ 10	1	9	32	3	4
クラスタ 11	15	2	4	8	3
クラスタ 12	4	3	6	9	1
クラスタ 13	9	2	11	1	3
クラスタ 14	5	4	2	7	6
クラスタ 15	3	4	1	2	1
クラスタ 16	2	1	2	2	1
クラスタ 17	4	–	3	1	–
クラスタ 18	2	–	4	1	–
クラスタ 19	1	–	4	–	–
クラスタ 20	4	–	–	–	–
クラスタ 21	1	–	–	–	–
クラスタ 22	1	–	–	–	–

最後に，式 (3.4) で表される無限関係モデルの関係パラメータ η_{kl} の分布を図 5.7 から図 5.11 に示す．横軸はクライアントのクラスタの番号，縦軸はサーバのクラスタの番号である．関係パラメータ η_{kl} はサーバのクラスタとクライアントのクラスタの間の関係の強さを表し，η_{kl} の値が大きいほどクラスタ間の関係が強い．無限関係モデルにより，関係が強いオブジェクト同士が同じクラスタに，関係が弱いオブジェクト同士が同じクラスタになるように分類されている．これを反映して黒 ($\eta_{kl} = 1$) に近いセル，白 ($\eta_{kl} = 0$) に近いセルが多い．

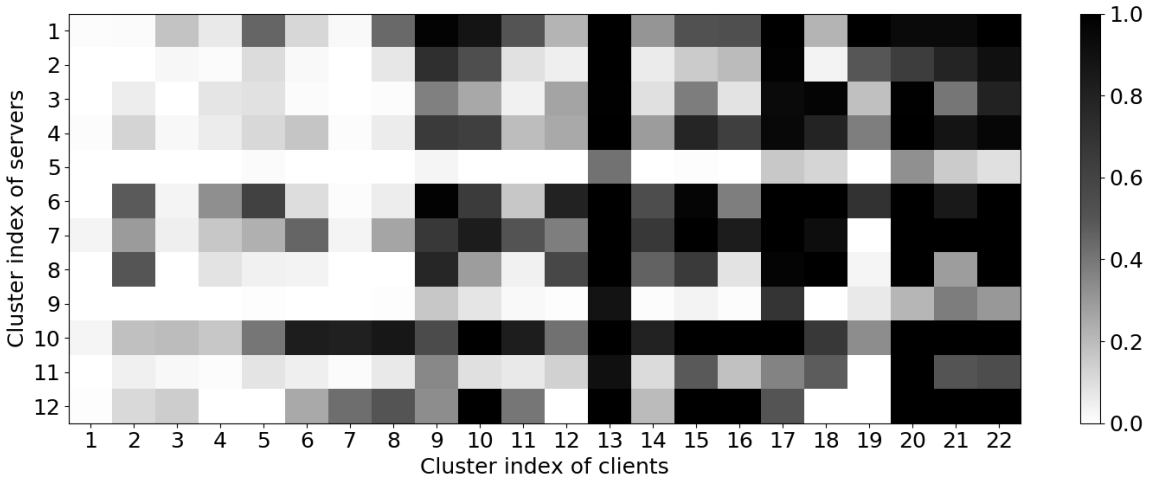


図 5.7: クラスタ間の関係パラメータ η_{kl} の分布 (期間グループ 1)

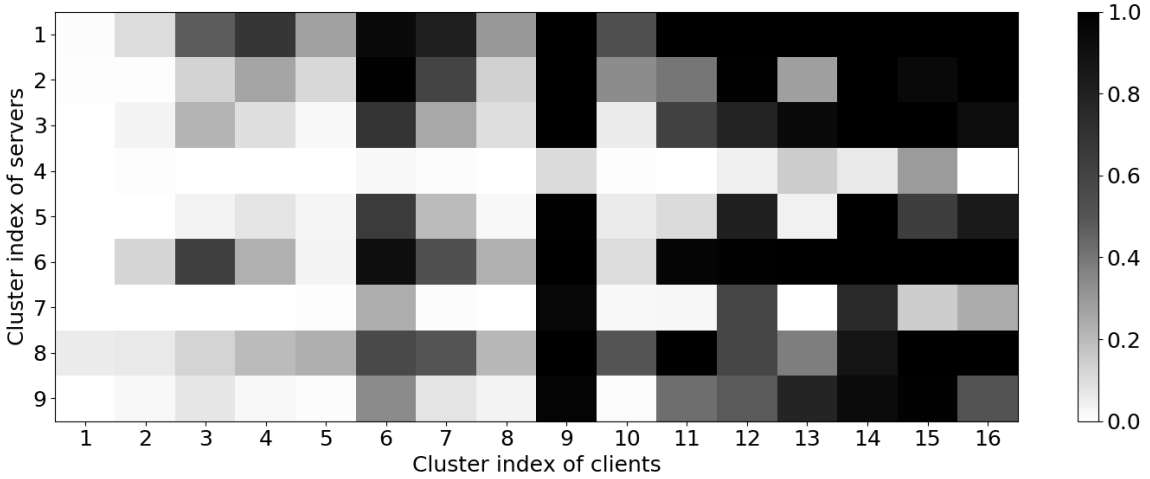


図 5.8: クラスタ間の関係パラメータ η_{kl} の分布 (期間グループ 2)

図 5.9: クラスタ間の関係パラメータ η_{kl} の分布 (期間グループ 3)

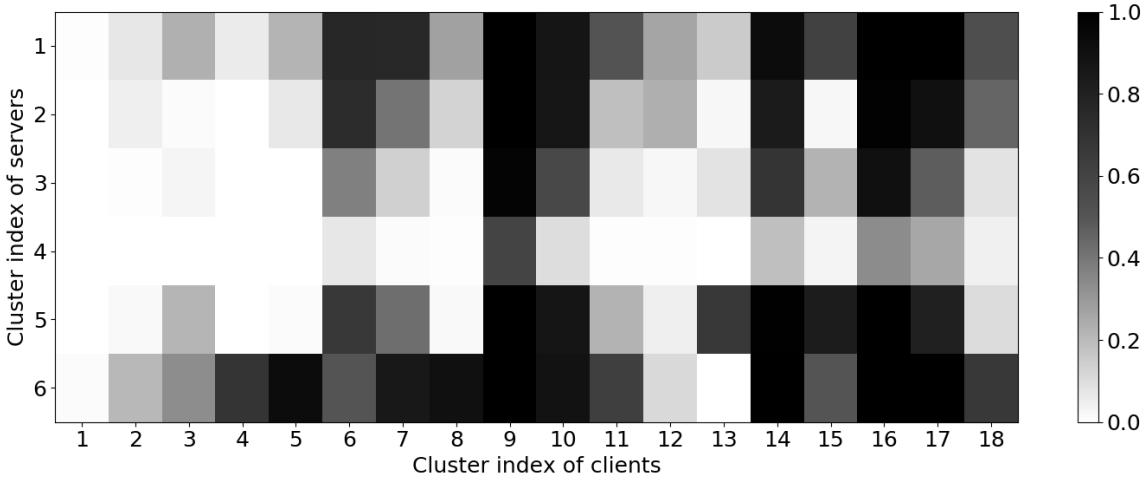


図 5.10: クラスタ間の関係パラメータ η_{kl} の分布 (期間グループ 4)

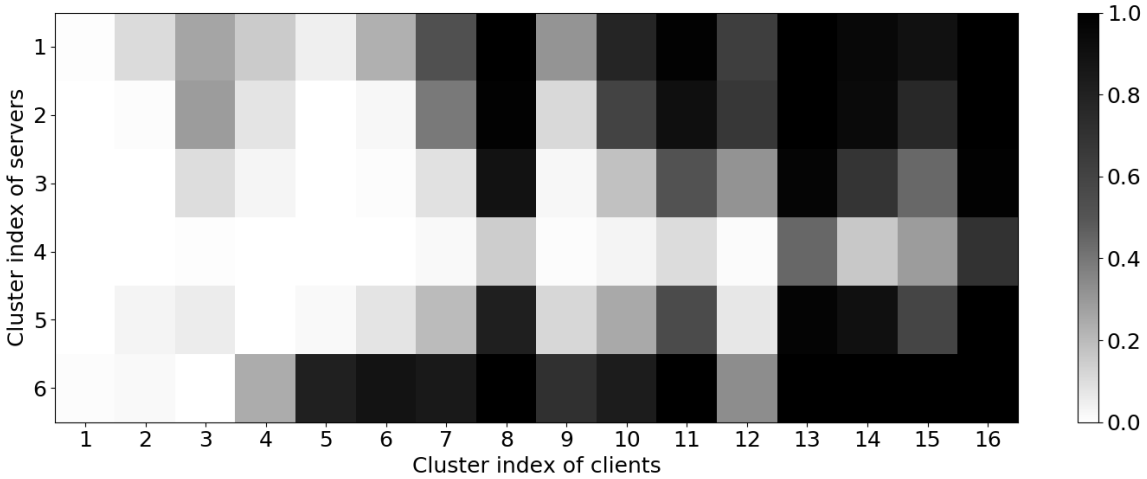


図 5.11: クラスタ間の関係パラメータ η_{kl} の分布 (期間グループ 5)

第 6 章

結論

6.1 まとめ

本研究では，共クラスタリングの一種である無限関係モデルを用いて HTTPS 通信の分類を行った．はじめに，x-means 法を用いて，クライアントとサーバの関係データが時間帯や平日・土日によって特徴が異なることを確認した．次に，無限関係モデルを用いることでクライアントとサーバの関係を保持したまま特徴を確認することができた．

このようにして，暗号化されている HTTPS 通信においても，暗号化されない情報を用いてトラヒックの特徴を分析することができる．

6.2 今後の課題

本研究で残された今後の課題を以下に述べる．

他の情報との組み合わせ

本研究では，ClientHello メッセージにおける送信元 IP アドレスと SNI の値を関係データ作成のために用いたが，他の情報を組み合わせるにより精度の向上が期待される．組み合わせる情報として，2.3 節で説明した暗号スイートのリストや，HTTPS 通信のパケットの個数やサイズなどが考えられる．

データ解析手法の変更

本研究では，データ解析手法として無限関係モデル (IRM) を用いた．しかし，4.2 節で述べたように IRM はスパースな関係データに対して適切にクラスタリングできない可能性があ

る．そこで，IRM の代わりに IRM の拡張手法である SIRM (Subset Infinite Relational Models) [24] を用いる．これにより，少数の重要なパターンのみ抽出できることが期待される．

謝辞

本修士論文の作成にあたり，日ごろよりご指導をいただいた早稲田大学基幹理工学研究科の後藤滋樹教授に深く感謝いたします．また，本研究を進めるにあたり，早稲田大学基幹理工学部情報理工学科の内田真人教授には的確なアドバイスをいただき大変感謝しております．最後に，日ごろよりご協力をいただきました後藤滋樹研究室の皆様に感謝いたします．

参考文献

[1] 佐藤 弘毅, 後藤 滋樹. “無限関係モデルを用いた HTTPS 通信の分類”. In: **電子情報通信学会総合大会講演論文集** 2019 (2019). (発表予定).

[2] Adrienne Porter Felt et al. “Measuring HTTPS adoption on the web”. In: *26th USENIX Security Symposium*. 2017, pp. 1323–1338.

[3] David Naylor et al. “The cost of the S in HTTPS”. In: *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*. ACM. 2014, pp. 133–140.

[4] *Official Google Webmaster Central Blog: Indexing HTTPS pages by default*. `https://webmasters.googleblog.com/2015/12/indexing-https-pages-by-default.html`

[5] *Chromium Blog: A secure web is here to stay*. `https://blog.chromium.org/2018/02/a-secure-web-is-here-to-stay.html`

[6] *Chromium Blog: Evolving Chrome's security indicators*. `https://blog.chromium.org/2018/05/evolving-chromes-security-indicators.html`

[7] *Let's Encrypt - Free SSL/TLS Certificates*. `https://letsencrypt.org/`

[8] *Service Worker API — MDN*. `httpshttps://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API`

[9] *HTTP Archive*. `https://httparchive.org/`

[10] Martin Husák et al. “Network-based HTTPS client identification using SSL/TLS fingerprinting”. In: *Availability, Reliability and Security (ARES), 2015 10th International Conference on*. IEEE. 2015, pp. 389–396.

[11] Charles Kemp et al. “Learning systems of concepts with an infinite relational model”. In: *AAAI*. 2006.

[12] E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.3*. RFC 8446. `https://www.ietf.org/rfc/rfc8446.txt`. Aug. 2018.

[13] T. Dierks and E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.2*. RFC 5246. `https://www.ietf.org/rfc/rfc5246.txt`. Aug. 2008.

[14] S. Blake-Wilson et al. *Transport Layer Security (TLS) Extensions*. RFC 3546. `https://www.ietf.org/rfc/rfc3546.txt`. June 2003.

[15] The Apache Software Foundation. *An In-Depth Discussion of Virtual Host Matching - Apache HTTP Server Version 2.4*. `https://httpd.apache.org/docs/current/en/vhosts/details.html`

[16] Wazen M Shbair et al. "Improving sni-based https security monitoring". In: *Distributed Computing Systems Workshops (ICDCSW), 2016 IEEE 36th International Conference on*. IEEE. 2016, pp. 72–77.

[17] Martin Husák et al. "HTTPS traffic analysis and client identification using passive SSL/TLS fingerprinting". In: *EURASIP Journal on Information Security* 2016.1 (2016).

[18] Anil K Jain. "Data clustering: 50 years beyond K-means". In: *Pattern recognition letters* 31.8 (2010), pp. 651–666.

[19] David Arthur and Sergei Vassilvitskii. "k-means++: The advantages of careful seeding". In: *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics. 2007, pp. 1027–1035.

[20] Dan Pelleg, Andrew W Moore, et al. "X-means: Extending k-means with efficient estimation of the number of clusters." In: *Icml*. 2000, pp. 727–734.

[21] 石岡恒憲. "クラスター数を自動決定する k-means アルゴリズムの拡張について". In: **応用統計学** 29.3 (2000), pp. 141–149.

[22] 石井 健一郎, 上田 修功. **続・わかりやすいパターン認識 - 教師なし学習入門** -. オーム社, Aug. 2014.

[23] Iku Ohama et al. "An Extension of the Infinite Relational Model Incorporating Interaction between Objects". In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer. 2013, pp. 147–159.

[24] Katsuhiko Ishiguro, Naonori Ueda, and Hiroshi Sawada. "Subset infinite relational models". In: *Artificial Intelligence and Statistics*. 2012, pp. 547–555.