MASTER'S DEGREE THESIS

Master of Science in Advanced Mathematics and Mathematical Engineering

Title: Service Flow Modelling in the Telecom Cloud

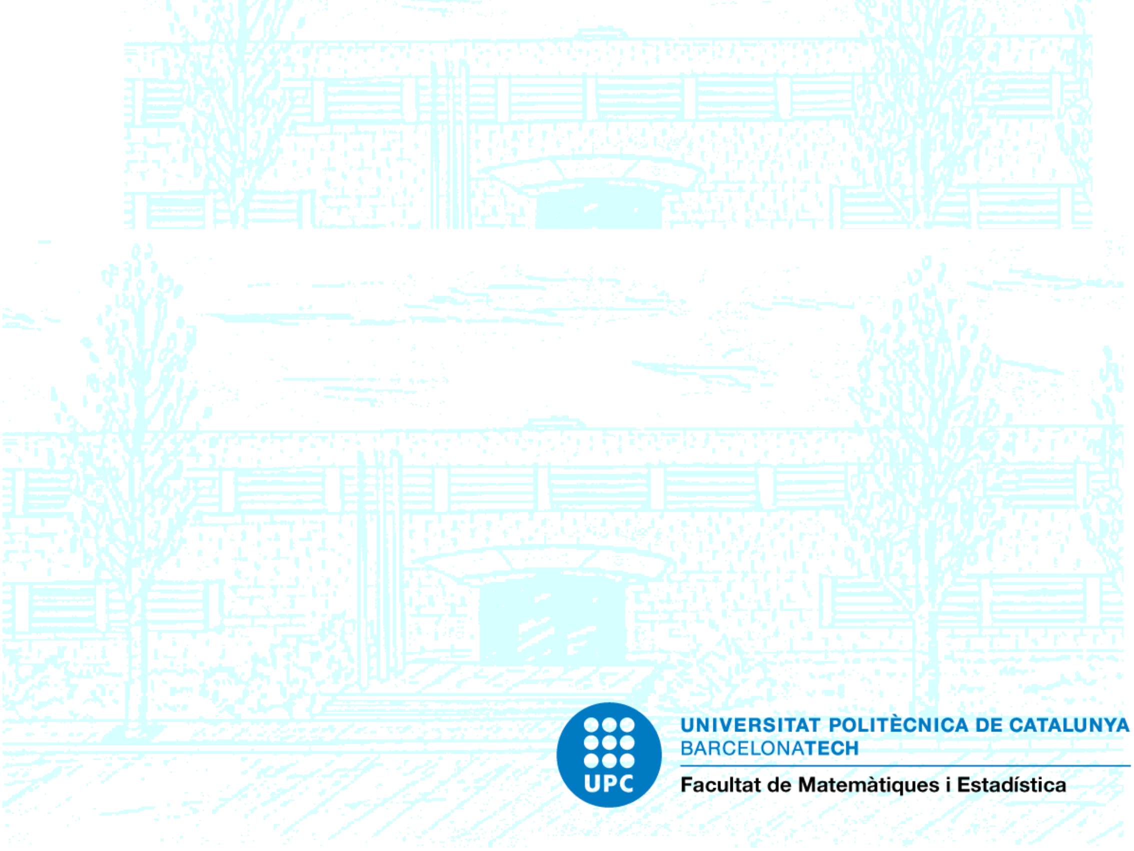
Author: Enle Lin

Advisor: Luis Velasco Esteban

Co-Advisor: Marc Ruiz Ramírez

Department: Computers Architecture

Academic year: 2015-2016



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat de Matemàtiques i Estadística

Abstract

Keywords: Service flow, Telecom Cloud, Big Data Analytics, statistical modelling.

MSC2010: 68U20

In telecom cloud infrastructures, a wide variety of network elements can be monitored to retrieve for many purposes, such as improving network performance and end user experience. Such wide and intense monitoring entails collecting huge volumes of data that needs to be transferred and stored, as well as being analyzed and fast processed to achieve near real-time performance. Therefore, Big Data techniques for data collection, pre-processing, and analysis and visualization have been recently proposed to provide a fully Big Data-backed ecosystem for telecom operators.

This project tackles the problem of service traffic flow modelling in the telecom cloud. A simulation and modelling procedure targeting at obtaining predictive models for realistic service traffic flows is developed. Distinct data analytics approaches can be emulated with the objective of evaluating the performance of distributed and centralized monitoring and modelling deployments.

Index

List of Figures

List of Tables

Chapter 1.

Introduction

1.1 Motivation and objectives

The increasing demand of new services such as Live-TV and Video on Demand (VoD) distribution is motivating a huge transformation of telecom operators. Due to the need to provide not only data transport but also computing services, they are deploying their own cloud infrastructure [Co12] to prove cloud services and enabling *Software Defined Networking (SDN)* [ONF16] and Network Functions Virtualization (NFV) [NFV]. The resulting infrastructure is referred to as the *telecom cloud* [Ve15]. NFV decouples network functions from proprietary hardware appliances, so they can be implemented in software and deployed on virtual machines (VM) running on commercial off-the-shelf computing hardware.

In telecom cloud infrastructures, a wide variety of network elements, servers and applications can be monitored to retrieve useful information for, among others, improving network performance and end user experience (e.g. see [Ru16]). Such wide and intense monitoring entails collecting huge *volumes* of data that needs to be transferred and stored assessing *validity*, as well as being analyzed and processed *fast* to achieve near real-time performance. Therefore, Big Data techniques for data collection, pre-processing, and analysis and visualization have been recently proposed to provide a fully Big Data-backed ecosystem for telecom operators [Gi16].

Recently, monitoring and modelling network traffic is receiving special attention due to its potential capacity to improve network performance. Specifically, proposed use cases including network reconfiguration based on future traffic estimation [Mo16] or prompt detection of traffic anomalies [AV16] are based on monitoring and modelling origin-destination (OD) traffic flows. An OD traffic flow can be defined as a stream of data packets between a source and a destination node (router). These kind of aggregated flows can be easily monitored since no distinction of services within the flow is required.

In case of requiring traffic models with finer granularity, e.g. service OD traffic models, traffic analysis tools such as Deep Packet Inspection (DPI) needs to be performed to monitor disaggregated traffic flows. DPI is a network function in charge of examining the data part (and possibly also the header) of a packet with the aim of searching for defined criteria to decide whether the packet may pass or if it needs to be routed to a different destination (e.g. for avoiding viruses spread, blocking attacks, or correcting protocol non-compliance) and collecting monitoring data of per service OD flows [Ro11].

DPI is also one of the most interesting use cases of NFV [Fi14]. Due to the large computing resources required for this exhaustive traffic analysis function, the distribution of virtualized DPI instances along the telecom cloud to achieve target performance needs to be studied, e.g. for collecting meaningful service traffic data to estimate predictive models. Note that how to collect that disaggregated service traffic data, as well as how to use data analytics procedures to maximize useful information extracted from that collected data is receiving recent research interest [Ma14].

This project tackles the problem of service traffic flow modelling in the telecom cloud. The contributions are two-fold: *i*) to develop a statistical modelling procedure targeting at obtaining service traffic prediction models from a heterogeneous set of input variables; and *ii*) to provide a simulation platform for generating realistic service traffic flows and evaluating the accuracy of the proposed modelling procedure for a wide range of monitoring architecture configurations. We assume that service traffic flow modelling can be done by means of DPI VNF instances deployed in telecom cloud datacenters.

In the context of this work, a predictive model is defined as a function that returns the expected traffic flow of a given service at a given time between a given OD node pair subject to a set of descriptive explanatory variables. Besides of the utility to predict traffic flows, models can also serve as tools for analyzing the significance of the explanatory variables and the relationship among them thus, serving as additional tools to better understand the behavior of service traffic flows and its impact on aggregated OD flows behavior.

The simulation platform is oriented to provide flexibility to configure different scenarios according to different monitoring architectures. To achieve this, two main blocks are clearly separated. The *OD flow generation block* provides simulated service and aggregated traffic flows according to the loaded network configuration and the characterization of different services. That monitoring data as well as available data related to the network is stored in separated data files for further model estimation. The *model fitting block* receives monitoring data and, after applying filtering and aggregation actions according to the configured monitoring architecture, runs a statistical procedure to obtain the model with the best trade-off between accuracy and number of coefficients (i.e. descriptive variables). Note that these modules can be independently executed, e.g. the model fitting block could be applied to real traffic traces obtained from monitoring real networks.

To facilitate even more the utility of the simulator and foster further improvements and enhancements, we have selected Matlab as software engine and platform. Matlab is optimized for solving engineering and scientific problems. A vast library of prebuilt toolboxes with basic and advanced algorithms is available. Good scalability, friendly programming interface, and easy integration with other languages and applications are some of the advantages behind our choice.

1.2 Report organization

The rest of the document is organized as follows. Chapter 2 approaches the necessary background to understand the contributions of this work. It starts with an introduction to the cloud-ready optical transport networks, then the typical service traffic in the network are briefly presented and finally, the Big data analytics architecture based on telecom cloud is explained.

In Chapter 3, the methodology for simulation platform is explained, some mathematical models are introduced to generate service traffic flows and the modelling procedures for this project are presented.

The contents of Chapter 4 are focused on the technical details of the implementation of the distinct modules in the simulator including: *i*) service traffic flow generation; *ii*) predictive model fitting, and *iii*) model validation.

Chapter 5 presents a case study for a reference scenario. The evaluation of the methodology for the scenario is concluded and the prediction models are presented and evaluated. Based on the models statistics, the comparison of modelling in different architectures is done.

Finally, Chapter 6 concludes the report with the main contributions and conclusions of the project.

Chapter 2.

Background

In this chapter, the necessary concepts are introduced in order to facilitate the understanding of the contents of this project. Firstly, the cloud-ready optical transport networks are presented, subsequently, the typical service traffic characterization and the basic concepts of DPI are presented. Finally, Big Data backed telecom cloud scheme for Big Data analytics architecture is explained.

2.1 Cloud-ready optical transport networks

An optical transport network can be defined as an undirected graph, where the edges are fiber optic links and the vertices are optical nodes, named Optical Cross Connects (OXC), capable of switching high-speed optical signals in a fiber optic network. The optical technology employs a range of frequencies of the total Optical Spectrum (OS), measured in Gigahertz (GHz). The capacity of an optical link depends on, among the others, the amplitude of OS.

On the top of described optical layer, large packet nodes (e.g., IP routers or Ethernet switches) are collocated with some OXCs and it serve as end points of network traffic, as well as to support intermediate transit routing/switching. Thus, an OD traffic flow represents an amount of data transported between an origin packet node and a destination packet node, usually expressed in Megabits per second (Mb/s) or Gigabits per second (Gb/s).

The transmission of OD traffic is supported by connections in the optical layer, called as *lightpaths*. From the abstracted view of the packet layer, a lightpath is considered as a virtual link directly connecting two packet nodes. Thus, a virtual topology is created and used to transmit OD traffic between origin and destination nodes.

Figure 2-1 presents a simplified network approach conceived for the understanding of the contents of this project. An optical transport network containing a set of packet nodes that are interconnected by means of virtual links at the packet layer is considered. Each virtual link is supported by one or more optical connections in the optical layer and OD traffic is served through such capacity. For the sake of simplicity, details on network connectivity are not depicted in the figure.

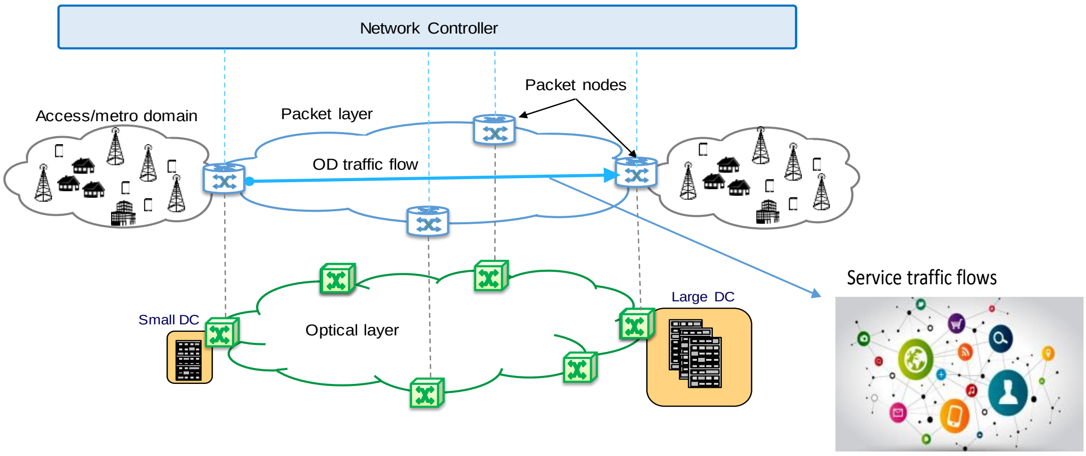


Figure 2-1: Considered network

The network in the example interconnects different metropolitan areas. All the service traffic generated in one area, such as mobile applications or data sharing, targeting other area in the operator's domain or another network (e.g., the Internet) are sent towards the destination node as an aggregated OD traffic flow, which contains all the service traffic flows.

Besides the aforementioned network approach, a set of datacenters of different sizes are integrated as part of the network infrastructure. These datacenters provide IT resources to, among others, support user services computational requirements and host virtual network functions (VNF). An example of VNF can be DPI used for analyzing OD traffic between metropolitan areas. Note that if a metropolitan area has no local computing resources (i.e. datacenter) to perform DPI function, its outgoing OD traffic should be sent to an intermediate destination where that function will be executed before reaching the final destination.

The considered cloud-ready transport network requires dynamic control of both network and computing resources. In fact, coordination between cloud and interconnection network is required to organize resources in both strata in a coherent manner, which is done by means of an intelligent network controller. Although no specific technology is strictly assumed for this control, the Application-Based Network Operations (ABNO) architecture proposed by the Internet Engineering Task Force (IETF) can be used as a centralized entity in charge of controlling the network in response to requests from the applications and services [RFC7491].

2.2 Service characterization

Service traffic flows generated at one metropolitan area belong to a wide variety: mobile applications, web browser, VoD, data migration, etc. In this project, these traffic flows are classified into four categories, namely: *Residential*, *Business*, *CDN* and *DC2DC*. Figure 2-2 shows the daily profiles of the services based on the definitions in references [SS] and [Mo16].

The main characteristics of these services are as follows:

- *Residential* traffic is generated by resident users from checking the weather or sports scores, shopping and banking, communicating with family and friends in myriad ways. This traffic is less in the morning and increases at night according to Figure 2-2.
- *Business* traffic is generated by employers for their work office, it includes fixed IP WAN or Internet traffic generated by businesses and governments. An average business user might generate 4 GB per month of Internet and WAN traffic, large-enterprise user would generate significantly more traffic, 8–10 GB per month [CISCO]. As show in the figure, it has two peaks: at the midday and 6:00 p.m., as the rush hours in the offices.
- A Content Delivery Network (*CDN*) is a system of strategically positioned servers around the globe, to avoid the latency problem because of long distance between origin server and users. These servers maintain copies of the content and are retrieved when a user looks up the website. The *CDN* can deliver images, HD video, 4K content, as well as a multitude of other files. The profile of this traffic starts increasing at 8:00 a.m. arriving the rush time at 6 p.m., and then decreases at 9 p.m.
- *DC2DC* (*Data Center to Data Center*) traffic is generated among datacenters by replication, back up, data migration, virtualization, and other Business Continuity/Disaster Recovery (BC/DR) flows. It has peaks in some hours when migration of data between datacenter occurs; that is, a huge volume of data traffic moving from one to the other datacenter. This huge volume of data will be transferred with a limit transmission speed until all the data is migrated. The transmission speed is slower at the day, when the network is saturated; and faster at night, when the network is more fluid.

Different mixes of these profiles lead to an extensive variety of aggregated OD traffic flows. To illustrate these differences, Figure 2-3 shows three examples of ODs between metropolitan areas of different sizes and different demand of services. Specifically, OD1 represents an example where DC2DC traffic clearly predominates, OD2 presents the case where residential and CDN are the most dominant services, and OD3 illustrates the case of similar proportion of four services. Note that not only the resultant aggregated daily pattern varies among examples but also traffic volumes ranges are different.

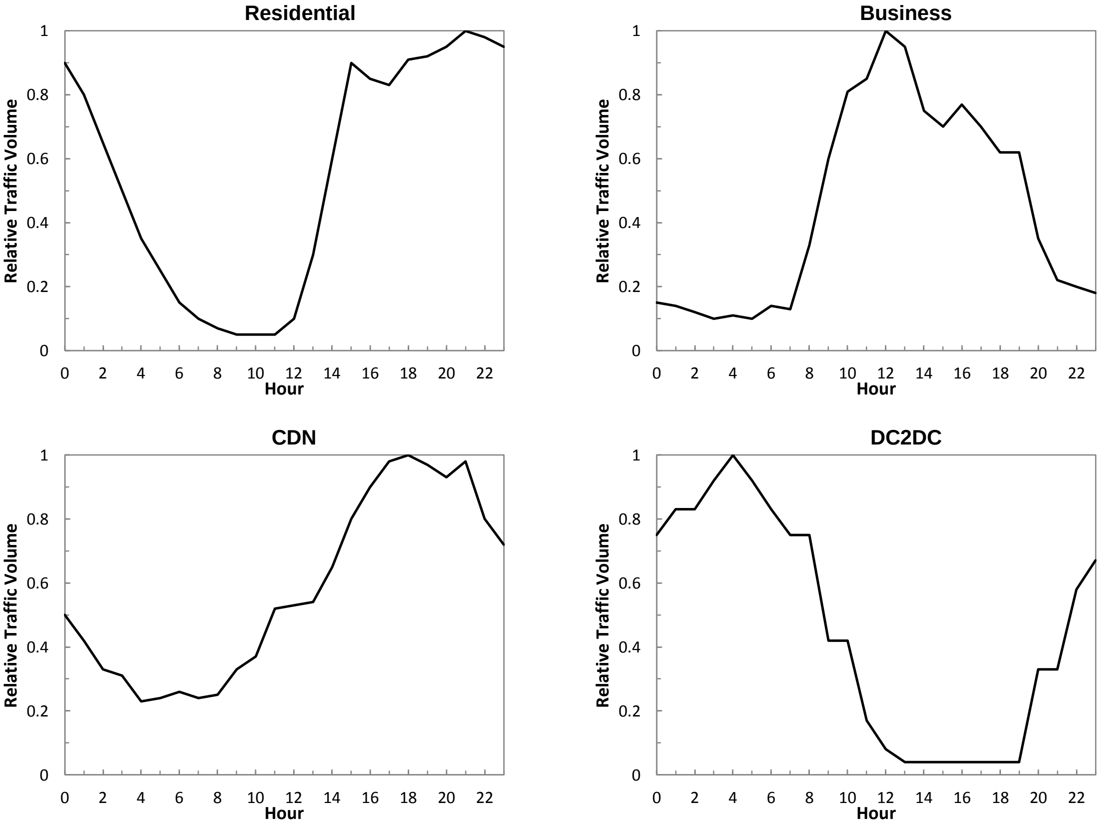


Figure 2-2: Service daily profiles.

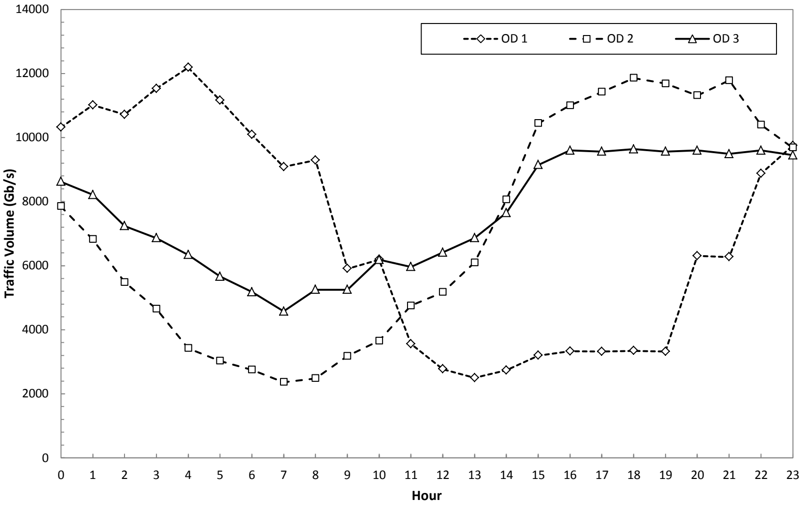


Figure 2-3: Example of aggregated OD flows

2.3 Big data analytics architecture

Traffic monitoring is an essential task for network operators since it allows evaluating network performance. To perform control upon the network, using data analytics in the observed traffic can be useful. For this purpose, data is recollected and appropriately stored, preprocessed, and modelled by predictive models that indicate the future evolution of the traffic. Figure 2-4 illustrates a general view of this approach.

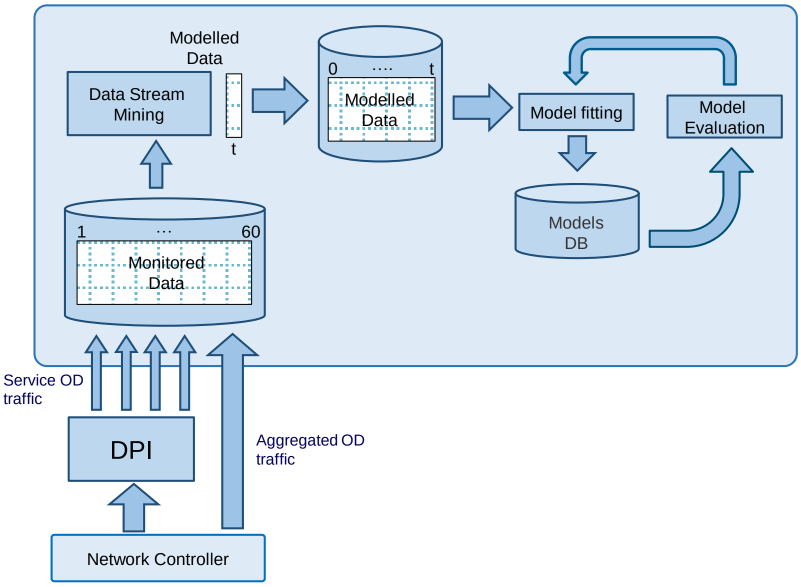


Figure 2-4: Big Data Backed telecom clouds scheme

Traffic is generated by users through different services as explained in the previous section. This generated OD traffic is transferred from origin node to the destination node as an aggregated flow. We assume that the traffic monitoring data is collected at the edge IP routers at regular intervals, e.g., every minute.

As previously introduced, a possible technique to extract information of the service flows within an OD flow is Deep Packet Inspection (DPI). DPI is a form of computer network packet filtering that examines the data part and the header of a packet as it passes an inspection point, searching for protocol non-compliance, viruses, spam, intrusions, or defined criteria to decide whether the packet may pass or if it needs to be routed to a different destination, or, for the purpose of collecting statistical information [Wi16.2]. DPI engines can be virtualized and dynamically deployed as pieces of software on commodity hardware [NFV]. In this work, we assume obtaining data from service traffic flows by means of virtualized DPI instances. That service traffic flow data is stored in a common data repository, named *Monitored Data Repository*.

Following a predefined time period, e.g. every hour, the collected data for a given OD pair in the *Monitored Data Repository* is summarized applying data stream mining, producing a *Modelled Data Repository*. This modelled data contains the minimum, maximum, average, the last value of each of the period, and a time stamp. Then, the *Model fitting* and *Model evaluation* are performed after a predefined number of modelled data periods.

In view of the architecture of the telecom cloud previously introduced in this chapter, two main schemes can be devised to implement the aforementioned Big Data analytics architecture in the telecom cloud:

- **Centralized**: similar to Figure 2-4, all monitoring data is collected in a unique repository, thus modelling cycle can be done with global view of the network. Regarding DPI instances placement, we can assume that those instances are either centralized or distributed closer to where the traffic is generated, thus avoiding sending extra traffic for inspection purposes. Although the latter allows reducing network traffic volume, it requires of distributed computational resources available for deploying DPI instances. Regardless of the DPI function deployment chosen, this architecture entails sending monitoring traffic data from a plenty of sources to the centralized repository.
- **Distributed**: data is stored and processed locally and as a consequence of this, predictive models have a local (partial) view of the network. This method allows reducing the impact of centralized databases (monitoring and modelled data) synchronization since they are stored in a distributed way. Moreover, distributed computational resources are better exploited; there is no need to have a large DC able to deal with data analytics from huge volumes of collected data. Note that this distributed approach can co-exist with the centralized one; e.g. independent traffic models can be computed in nodes with the data monitored and inspected locally, whereas those models can be synchronized in a centralized repository for a deeper correlated analysis aiming at improving the accuracy of traffic models.

2.4 MATLAB simulation environment

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment with a proprietary programming language (M programming language) developed by MathWorks. The main features of MATLAB are, among others: matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, Java, FORTRAN and Python.

Throughout this project, the traffic generation and modelling are implemented and embedded in an MATLAB based simulator. The simulator is organized in two blocks: *i) flow generation block* to emulate the real performance of an optical network and *ii) model fitting block* to predict the future traffic in this network.

The *flow generation block* has been implemented to simulate several functionalities. Its combination allows emulating the protocols of a network. Different optical network topologies can be created specifying a configuration of nodes.

Each node is able to generate traffic according to the mathematical models which are described in Chapter 4. The traffic prediction module is also explained in that chapter, which is in charge of analyzing the provided traffic and node information to build from it models that are later used to predict the network traffic.

2.5 Summary

In this Chapter, the basic concepts of the telecom cloud focusing on the Big Data architecture for service traffic analysis were presented to understand the aim of this project. Based on these basic concepts, next chapters are devoted to present the main contributions of this project.

Chapter 3.

Methodology

In this chapter, we introduce the mathematical models and procedures used for service traffic flows generation, fitting predictive models and validation. It starts with the notation of the set and the parameters used in the models formulation. Then, the mathematical models for the service flow are presented, followed by the explanation of the prediction and validation methods. Finally, the proposed algorithms for modelling and validation are explained.

3.1 Notation

The following sets and parameters are necessary to explain the content of this chapter:

$G(N,L)$ Network graph, where N represents the set of nodes in the network and L the links between nodes.

S Set of traffic services, index s.

OD Set of OD pairs. Every element in OD contains a tuple $<i, j>$ of nodes indicating the origin and destination of such OD, i.e. $i \rightarrow j$

OD_s Set of OD pairs which have traffic flow of service s.

t absolute time in the simulation.

T period for the service profiles.

τ relative time in the simulation, defined as mod(t,T).

P set of explanatory variables, index p.

$Y_{ijs}(t)$ traffic flow of service s from node i to j in the time step t.

We consider that the period T is a known variable, so, τ can be deducted with t. Thus, the use of t and τ will be indifferent to our consideration.

The object of traffic generation and prediction is Y_{ijs}, which is a time series characterized by the source i and destination j nodes of an OD and a traffic service. In other words, Y_{ijs} represents a specific service traffic flow.

3.2 Service traffic generation

In this section, we explain some models to generate OD traffic flows of the services that we considered in this project. Note that these models can be used as modules to compose more complex service models.

3.2.1 Profile model

A simple way to model service traffic is to use only their average profiles. The formulation for this model is:

$$Y_{ijs}(t) = f_s(t), \tag{3.1}$$

where $f_s(t)$ is the profile function of the service s. For simplicity, all the service profiles are considered to be periodic with period T. Then, the profile function can be defined by interpolating a set of profile values in the period, $I = \{\varphi(0), \varphi(T)\} \cup \{\varphi(t_h) \mid t_h \in (0,T)\}$. The formulation for $f_s(t)$ is, using the relative time, τ, instead of absolute time, t,

$$f_s(\tau) = \begin{cases} \varphi(a)\left(1 - \dfrac{\tau - a}{b - a}\right) + \varphi(b)\left(\dfrac{\tau - a}{b - a}\right), \quad \tau_h \le \tau \le \tau_{h+1}, \quad \varphi(a), \varphi(b) \in I \end{cases}. \tag{3.2}$$

In order to fit a more realistic case, the service traffic flow presents a random variation from the model, so a random number with normal distribution is added as variate term:

$$N(0, \sigma\ (f_s(\tau))). \tag{3.3}$$

Therefore, the variance of the normal distribution depends on the profile value, which is reasonable since the services present higher deviation as more traffic flow is generated.

The resulting model is:

$$Y_{ijs}(t) = \left(f_s(\tau) + N(0, \sigma\ (f_s(\tau)))\right). \tag{3.4}$$

3.2.2 Weighted model

Besides the profiles, the service generation models could depend on some weight terms in the origin nodes, w_i

$$Y_{ijs}(t) = w_i \cdot f_s(\tau), \tag{3.5}$$

or on the weight terms in the destination nodes, w_j

$$Y_{ijs}(t) = w_j \cdot f_s(\tau), \tag{3.6}$$

or on the terms in both nodes, addictive or multiplicative:

$$Y_{ijs}(t) = (w_i + w_j) \cdot f_s(\tau) \quad , \qquad Y_{ijs}(t) = w_i \cdot w_j \cdot f_s(\tau). \tag{3.7}$$

In this model, the service traffic flow depends on two independent parameters of both nodes and the service profile. As in the profile model, it has a variate term to be realistic. The final formulation for weighted model is:

$$Y_{ijs}(t) = (w_i + w_j) \cdot \left(f_s(\tau) + N\left(0, \sigma\left(f_s(\tau)\right)\right)\right), \quad Y_{ijs}(t) = w_i \cdot w_j \cdot \left(f_s(\tau) + N\left(0, \sigma\left(f_s(\tau)\right)\right)\right), \tag{3.8}$$

where w_i or w_j may be 1 in the case that only depends on the weights of one node.

3.2.3 Gravity models

Gravity models are a type of models, developed largely in the social sciences, for describing aggregate levels of interaction among the people of different population. They have traditionally been used mostly in areas such as geography, economics, and sociology, but also have found applications in the network traffic analysis [Ko14].

The term gravity model derives from the fact that, in analogy to Newton's law of universal gravitation, it is assumed that the interaction among two populations varies in direct proportion to their size, and inversely, with some measure of their separation, e.g. the distance.

The expected traffic represents the prediction of the flow created by two nodes and calculated by the gravity equation [Gr03]:

$$w_{ij} = k_{ij} \frac{m_i \cdot m_j}{d_{ij}}, \tag{3.9}$$

where m_i and m_j are the mass term in both nodes and the d_{ij} is the measure of their separation. Note that this model depends on a term dependent on both origin and destination nodes.

Then, we introduce the gravity model for the service traffic generation as:

$$Y_{ijs}(t) = w_{ij} \cdot f_s(\tau) . \tag{3.10}$$

Adding the variate term as the previous models:

$$Y_{ijs}(t) = w_{ij} \cdot \left(f_s(\tau) + N\left(0, \sigma\ (f_s(\tau))\right)\right). \tag{3.11}$$

3.2.4 Evolutionary model

For services that presents some extra traffic flow that does not depend on the profile, it can be modeled with an additive function, *g(t,P)*, so, these services can be modeled with the following formulation:

$$Y_{ijs}(t) = \tilde{Y}_{ijs}(t) + g(t, P) , \tag{3.12}$$

where $\tilde{Y}_{ijs}(t)$ indicates the service traffic that depends on the profile, which can be modeled with previous models; and *g(t,P)* is the extra flow which depends on time and other variables in *P* (number of population, migration of a datacenter, etc.).

3.2.5 Dependent model

Finally, some service traffic could depend on the flows of same service generated by OD pair with same origin node:

$$Y_{ijs}(t) = \sum_{\substack{ij' \in OD_s \\ j' \neq j}} c_{ij'} Y_{ij's}(t) , \tag{3.13}$$

where $c_{ij'}$ are scalars; and $Y_{ij's}$ are the traffic flows of service *s* generate by pair $ij' \in OD_s$. Another version of this model formulation is that the service traffic could depends on the flow generated OD pairs with same destination node:

$$Y_{ijs}(t) = \sum_{\substack{i'j \in OD_s \\ i' \neq i}} c_{i'j} Y_{i'js}(t) . \tag{3.14}$$

3.3 Service traffic prediction

Time series analysis and linear regression are models that can be used to predict values of a time series. In this section, these two modelling methods will be presented.

3.3.1 Time series analysis

A time series is an ordered sequence of values of a variable at equally spaced time intervals over a continuous time interval.

Time series analysis contains methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. A time series model predicts future values based on previously observed values [We94].

The *autocorrelation* refers to the correlation of a time series with its own past and future values. Let *Y(t)* = $\{y_1,\ldots,y_n\}$ be a time series and *Y(t+k)* = $\{y_{k+1},\ldots,y_{n+k}\}$ the same time series lagged by *k* time units (k ≥ 0); then the autocorrelation between Y_t and Y_{t+k} is given by autocorrelation function (ACF)

$$ACF = \frac{Cor(Y(t), Y(t+k))}{\sigma(Y(t)) \cdot \sigma(Y(t+k))}, \tag{3.15}$$

where *Cor* is the covariance function and *σ* is the standard deviation.

Positive autocorrelation might be considered a specific form of "persistence", a tendency for a system to remain in the same state from one observation to the next. For example, the likelihood of tomorrow being rainy is greater if today is rainy rather than if today is dry.

Given a time series *Y(t)*, the *partial autocorrelation* of lag *k*, denoted *a(k)*, is the autocorrelation between *Y(t)* and *Y(t+k)* that is not accounted for by lags *1* to *k−1*, inclusive.

$$\alpha(1) = Cor\big(Y(t+1), Y(t)\big) \ , \tag{3.16}$$

$$\alpha(k) = Cor\big(Y(t+k) - P_{t,k}\big(Y(t+k)\big), Y(t) - P_{t,k}\big(Y(t)\big)\big) \qquad \text{for } k \geq 2 \ , \tag{3.17}$$

where $P_{t,k}(Y)$ denotes the projection of *Y* onto the space spanned by $Y_{t+1},\ldots,Y_{t+k-1}$.

Some time series data have presence of sparse sampling, that is, the intervals between time points are not uniform in general [Jh15]. In this case, the associate temporal series model will get false autocorrelation due to the sparsity of observations.

For example, a equidistant time series of size 10000 generated by

$$Y(t) = c + 0.5 \cdot Y(t-1) + \varepsilon(t) \ , \tag{3.18}$$

where *c* is a scalar and *ε(t)* is a white noise process with zero mean and constant variance. In the Figure 3-1 shows the autocorrelations of this time series, one can observe that for lag greater than 7, the values of ACF belong to [-*0.2,0.2*] (the blue lines); and for lag greater than 1, the values of partial ACF belong to same interval. After randomly deleting 5000 observations, the new autocorrelations are computed (See Figure 3-1), and one can observe that there are autocorrelations for large lag values, which are false since the time series is generated only with the value of the previous time step, so there are no autocorrelation in the far time steps. In view of the problem that presents time series analysis with sparse dataset, another prediction method will be considered: the linear regression models.

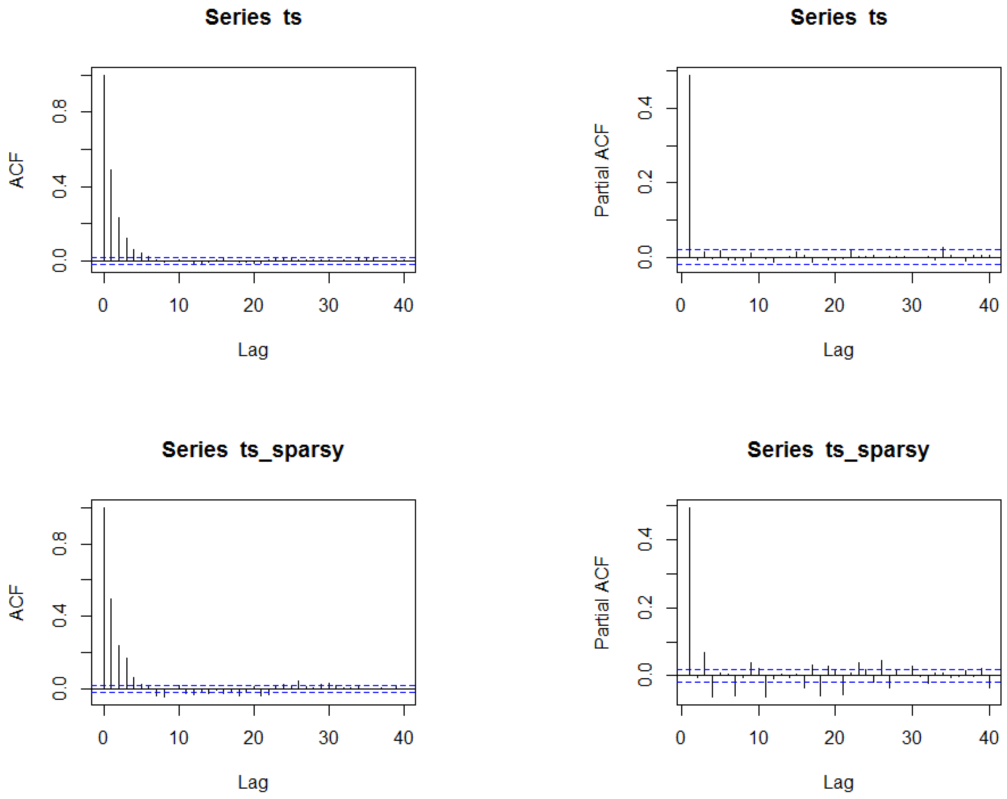


Figure 3-1: Illustration of generated data of an OD pair.

3.3.2 Linear regression

Regression analysis are statistical processes for estimating the relationships among variables. Linear regression was the first type of regression analysis to be studied rigorously and to be used extensively in practical applications.

Linear regression is an approach for modeling the linear relationship between a scalar dependent variable Y and one or more explanatory variables (or independent variables) denoted X_p. The relationships are modeled using linear predictor functions whose unknown model coefficients are estimated from the data. Given a data set $\{Y,X_1,\ldots,X_p,\ldots\}$, the linear regression model take the form

$$Y = \sum_{p \in P} C_p \cdot X_p + C_0 + \varepsilon \quad , \tag{3.19}$$

where c_p denotes the coefficients for variable x_p; c_0 is the intercept term, which is a scalar; and ε is an error variable.

If the response variable is a time series, *Y(t)*, then some time dependent predictive variables can be considered [Ra95]:

$$Y(t) = \sum_{p' \in P} B_{p'} \cdot W_{p'}(t) + \sum_{p \in P} C_p \cdot X_p + C_0 + \varepsilon(t) \quad , \tag{3.20}$$

where $B_{p'}$ are the coefficient for $W_{p'}(t)$.

An extension formulation can include also explanatory variable values in previous time steps:

$$Y(t) = \sum_{p' \in P} \sum_{t' \le t} B_{p't'} \cdot W_{p'}(t') + \sum_{p \in P} C_p \cdot X_p + C_0 + \varepsilon(t) \quad . \tag{3.21}$$

Also the values of the response variable in the previous time steps ($t' < t$ since the *Y(t)* is the value which have to predict) can be considered:

$$Y(t) = \sum_{\hat{t} < t} A_{\hat{t}} Y(\hat{t}) + \sum_{p' \in P} \sum_{t' \le t} B_{p't'} \cdot W_{p'}(t') + \sum_{p \in P} C_p \cdot X_p + C_0 + \varepsilon(t) \quad . \tag{3.22}$$

With the notation of Y_{ijs}, the formulation become:

$$Y_{ijs}(t) = \sum_{\hat{t} < t} A_{\hat{t}} Y_{ijs}(\hat{t}) + \sum_{p' \in P} \sum_{t' \le t} B_{ijsp't'} \cdot W_{ijsp'}(t') + \sum_{p \in P} C_{ijsp} \cdot X_{ijsp} + C_0 + \varepsilon_{ijs}(t) \tag{3.23}$$

Knowing this, we can also construct a predictive model for time series with linear regression. And it has no problem with sparsity since we could model without the values of the response variable at previous time steps.

Linear regression predicts the expected value of a response variable as a linear combination of a set of observed explanatory variables values. This implies that a constant change in an explanatory variable leads to a constant change in the response variable. This is appropriate when the response variable has a normal distribution (data that only varies by a relatively small amount in each direction, e.g. human heights). However, these assumptions are inappropriate for some types of response variables [Ne72]. For example, in cases where the response variable is expected to be always positive and varying over a wide range, e.g. income salary; or cases when the model predicts the probability that an event occurs.

A solution for non-normal distributed data is do a power transformation in the response variable to get it normal-like. The Box-Cox transformation is a good approach, it transforms no normally distributed data to a set of data that approximates to normal distribution.

The Box-Cox transformation of the variable Y is also indexed by parameter λ, and is defined as

$$Y'_{\lambda} = \frac{Y_{\lambda} - 1}{\lambda} \quad , \tag{3.24}$$

If $\lambda = 0$, then

$$Y'_{\lambda} = \log(Y_{\lambda}) \quad , \tag{3.25}$$

The algorithm calls for finding the λ value that maximizes the Log-Likelihood Function.

3.4 Validation

For the validation of the models, the AIC information and the R-squared (coefficient of determination) are appropriate statistics to measure the goodness of the models.

3.4.1 Akaike Information Criterion (AIC)

AIC is founded on information theory; it's a measure of the relative quality of the models for a given set of data. It deals with the balance between the model's goodness of fit and the complexity of the model. AIC does not provide a test about the quality of the model in an absolute sense. If all candidate models fit poorly, AIC will not give any warning.

The AIC information of a model is calculated with following formula [Ah14]:

$$AIC = 2k - 2\ln(L) \quad , \tag{3.26}$$

where L is the maximum value of the likelihood function for the model and k is the number of estimated parameters in the model.

Given a set of candidate models for the data, the preferred model is the one with the minimum AIC value. Hence AIC rewards goodness of fit (as assessed by the likelihood function), but it also includes a penalty that is an increasing function of estimated parameters' number.

3.4.2 R-squared

The R-squared, also called the coefficient of determination, is a number that indicates the proportion of the variance in the dependent variable, which is predictable from the independent variable for a model [Wi16].

If $Y_1, Y_2, \dots, Y_n$ is a vector of dependent variable of size n and $Z_1, Z_2, \dots, Z_n$ for their respective predicted values. The mean of the observed data is

$$\bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i \quad . \tag{3.27}$$

The total sum of square is

$$SS_{tot} = \sum_i (Y_i - \bar{Y})^2 \quad . \tag{3.28}$$

The regression sum of squares is

$$SS_{reg} = \sum_i (Z_i - \bar{Y})^2 \quad . \tag{3.29}$$

The sum of residuals square is

$$SS_{res} = \sum_i (Y_i - Z_i)^2 \quad . \tag{3.30}$$

A general version, based on comparing the variability of the estimated errors with the variability of the original values, is

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad , \tag{3.31}$$

Another version is common in statistics texts but holds only if the modeled values are obtained by ordinary least squares regression, which must include a fitted intercept or constant term.

The coefficient of determination is an important statistical measure of how well the regression line approximates the real data points. A value 1 for R-squared indicates that the regression line perfectly fit the data.

3.4.3 Tests for residuals

Although a high R-squared value is an important asset of the linear regression, there are objections to relying exclusively on this empirical criterion [Wi99]. Then, it is recommended to process a complement analysis of residual to validate the model. Two assumptions for validation of regression models are that the residuals are normally distributed and independents.

The *Jarque-Bera test* is a test decision for the null hypothesis that the data in a variable X comes from a normal distribution. The alternative hypothesis is that it does not come from such a distribution. The result h is 1 if the test rejects the null hypothesis at the 5% significance level, and 0 otherwise.

The test statistic is

$$JB = \frac{n}{6}\left(s^2 + \frac{(k-3)^2}{3} \right), \tag{3.32}$$

Where *n* is the sample size, *s* is the sample skewness, and *k* is the sample kurtosis. If the data comes from a normal distribution, the Jarque-Bera statistic asymptotically has a chi-squared distribution with two degrees of freedom, so the statistic can be used to test the hypothesis that the data are from a normal distribution.

The *Ljung-Box Q-test* is a test that assesses the null hypothesis that a series of residuals exhibits no autocorrelation for a fixed number of lags *L* at 5% significate level, against the alternative that some autocorrelation coefficient $\rho(k)$, $k = 1,...,L$, is nonzero [Ts10].

The test statistic is

$$Q = n(n+2)\left(\frac{\rho(k)^2}{(n-k)} \right), \tag{3.33}$$

Where *n* is the sample size, *L* is the number of autocorrelation lags, and $\rho(k)$ is the sample autocorrelation at lag *k*. Under the null hypothesis, the asymptotic distribution of *Q* is chi-square with *L* degrees of freedom.

3.4.4 Error computation

A measurement to evaluate the fitting of the models could be the mean prediction error at all the OD pairs, calculated with the following formula:

$$E = \frac{\sum | a_{ijs} - b_{ijs} |}{N_s} \tag{3.34}$$

Where a_{ijs} denotes real flow from location *i* to location *j* for service *s*; b_{ijs} denote the same flow predict by the models; N_s is the number of pair of OD that interchange flow of service *s*. The most suitable model is the one which minimizes this error.

3.5 Proposed algorithms

In this section, some proposed algorithms for modelling and validation are explained, each of them using previously detailed modelling and validation procedures.

3.5.1 Stepwise regression

Stepwise regression is a systematic method for adding and removing terms from a linear model, based on their statistical significance in explaining the response variable. There are different criteria to measure the statistical significance, such as AIC or R-squared.

The Step 3: The method searches for variables in *P'* to remove from the model according to the criteria (lines 14-21). If some variable is removed, go back to step 2; otherwise, end.

The algorithm finishes when the set *P'* contains the most appropriate variables to fit the model, according to the selected criteria. Besides to select linear terms of the available variable in *P*, the algorithm can also select cross-product terms or higher order terms of a variable to find the best set of explanatory variables according to the selected criteria.

Algorithm 3-1 presents the stepwise regression process. The method starts with no explanatory variables in the stepwise model (lines 2-3) and then enters or removes one of them based on the selected criteria. The main steps are:

Step 1: Fit the initial model.

Step 2: The method searches for variables in *P* to add to *P'* (set of explanatory variables to fit the model) according to the criteria, and repeat this step until no more variables can be added (lines 7-13). Goodness-of-Fit (GoF) of models is computed to decide whether an incremental model improves the best model obtained so far.

Algorithm 3-1: Stepwise regression algorithm.

```
Procedure Stepwise regression Algorithm
Input: y, P, data
Output: model
1:  begin
2:      P'=Ø
3:      model=lm(y,Ø)
4:      conv=False
5:      while not conv
6:          conv=True
7:          for x in P\P' do
8:              m=lm(y~P'∪x, data)
9:              if GoF(m)> GoF(model) then
10:                 model=m
11:                 P'=P'∪x
12:             endif
13:         end
14:         for x in P' do
```

```
15:         m=lm(y~P’\ {x}, data)
16:         if GoF(m)> GoF(model) then
17:           model=m
18:           conv=False
19:           break
20:         endif
21:       end
22:     end
23: end
```

Step 3: The method searches for variables in *P’* to remove from the model according to the criteria (lines 14-21). If some variable is removed, go back to step 2; otherwise, end.

The algorithm finishes when the set *P’* contains the most appropriate variables to fit the model, according to the selected criteria. Besides to select linear terms of the available variable in *P*, the algorithm can also select cross-product terms or higher order terms of a variable to find the best set of explanatory variables according to the selected criteria.

3.5.2 Transformation of variables

In linear regression models, there is a set of available variables that can be used for the modelling. However, some derived terms of these variables can be created in order to obtain more linearly correlated with the response variables. We process an analysis of the variables to create high order terms (square, cube, etc.) using both AIC information and R-squared measurement. The process is described in Algorithm 3-2, it starts with generation of traffic data for *n* days, then, for each explanatory variable *X*, it process the following steps:

Step 1: Build the linear models with power terms of *X* (line 7)

Step 2: Compute R-squared values and AIC information for the linear models. (lines 6-10)

Step 3: Find, from the set of linear models with highest values of R-squared, the model with minimum AIC information. (lines 14-21)

The most suitable model is which has highest R-squared value and lowest AIC according to the theory. With this in mind, first, it find the highest R-squared value, *R_high* (line 11); then establishes set of linear models with those that have R-squared value greater than *R_high* – δ, δ small; and find the model in the set, which has the lowest AIC. All the powers terms used in this model are considered available explanatory variables.

Adding power terms of variables that are not related with response ones do not improve the model fitting. Then, instead of applying this algorithm exhaustively to all the explanatory variables, some previous analysis to extract the relationships between the explanatory variables and response ones are done, in order to select appropriate variables to apply it.

Algorithm 3-2: Transformation of variables.

```
Procedure Transformation of variables
Input: data, P, R, data
Output: P′
Begin
  Models = Ø
  R_Models = Ø
  P’ = P
   for x in P, y in R
     for i := 1 to m do
           model = stepwise_regression(y,{$x,x^2,…,x^i$}, data)
           r(i) = r_squared (model)
           AIC(i) = AIC(model)
     end
     R_high = max(r)
     term_to_add = find(r == R_high)
     AIC_min = AIC(term_to_add)
     for i := 1 to m do
       if (r(i) < R_high - δ)
         if (AIC(i) < AIC_min)
           term_to_add = i
           AIC_min = AIC(i)
         endif
       endif
     end
     n = term_to_add
     $P’=P’\cup\{x,x^2,…,x^n\}$
   end
end
```

3.5.3 Selection of candidate models

For the modelling, consider that there is not only one model for the prediction but a set of candidate models. Algorithm 3-3 explains the candidate models selection’s process: the simulator builds new candidate models during the first d_1 days (line 2-4) and establishes Models as the set of candidate models; subsequently, it does the traffic flows prediction with Models and calculates the prediction errors with formulation in 3.35 (9-10). Then, removes the one with largest error remaining only d_1-*1* models (line 11-17). Apart from this, it introduces a new candidate model to the set at intervals of d_2 days, $d_2 > d_1$, and the model with the largest error is removed from Models when a new model is introduced.

Algorithm 3-3: Selection of candidate models.

```
Procedure Selection of candidate models
Input: y, P, day, Models
Output: Models
begin
  if day <= $d_1$ or mod(day,$d_2$) == 0 then
    model = stepwise_regression(y, P)
    Models = Models ∪ model
    if day == $d_1$ or mod(day,$d_2$) == 0 then
      remove = True
    endif
  endif
predict(Models)
calculate errors for model in Models
  if remove then
    if (error(model) == error(Gof(Models)) then
      Models = Models\model
      remove = False
    endif
  endif
end
```

3.6 Summary

In this chapter, the methodology for the data generation, the modelling and the validation has been presented. The 5 traffic generation models exposed in this chapter will be used to generate the service flows introduced in the previous chapter. The time series analysis, linear regression and the validation methods will be considered for the modelling procedure. Finally, some proposed algorithm are presented to enhance the fitting and validation of service traffic flow models.

In the following chapter, technical details about the implementation of the different techniques into the simulator platform will be presented.

Chapter 4.

Simulator

This chapter is devoted to the design and description of the simulation platform. Specifically, after introducing a global overview of the simulator, it tackles theoretical sections 3.2, 3.3 and 3.4 of the previous chapter from a practical perspective. A Matlab-based implementation of the simulator has been developed during this project following the specifications in this chapter

4.1 Design

The simulator basically consists in code blocks (scripts and functions) and databases (DB) composed to create two main blocks clearly separated:

- The *OD traffic generation block*: provides simulated service and aggregated traffic flows according to the loaded network configuration and the characterization of different services. That monitoring data as well as available data related to the network is stored in separated data files for further model estimation.
- The *model fitting block* receives monitoring data and, after applying filtering and aggregation actions according to the configured monitoring architecture, runs model estimation procedures to obtain the model with the best trade-off between accuracy and number of coefficients (i.e. descriptive variables).

The Figure 4-1 illustrates the OD traffic generation block of the simulator. It requires network configuration data as input (see next subsection for details about network configuration parameters). Part of this configuration data is used to generate service traffic flows between every of the (allowable) network nodes according to the user defined models. At each generation timer (e.g. every minute), a traffic sample for each of the service flows for all the OD pairs following the service profiles is generated and stored in the Raw traffic flow DB. This DB contains a quite unrealistic high level of traffic detail. For this reason, with a longer period that the generation timer (e.g. every 15 minutes), the aggregation timer triggers when average traffic values need to be computed. This aggregated-in-time traffic is then replicated and sent to two distinct DBs: the *Per-service OD flows* DB where traffic is stored as it is received, and the *Aggregated OD flows DB* where the sum of all the service flows in the same OD is firstly performed. With this procedure, the former DB contains traffic similar to that that could be monitored by means of some traffic analysis tool such as DPI, whereas the latter could correspond to the aggregated traffic monitoring performed by measuring traffic bitrate e.g. at router interfaces.

The model fitting block of the simulator is detailed in Figure 4-2. It requires of the monitoring traffic DBs (i.e. those created in the generation part) and also the network configuration parameters DB. Before starting modelling, data coming from that DBs needs to be filtered according to different criteria. Regarding network configuration, some data used to generate service traffic profiles is hidden in order to perform a fair evaluation of model fitting procedures. On the other hand, the selected monitoring configuration will affect the amount and type of data that can be used from the per-service OD flows DB. After applying those filters, a model fitting DB is generated combining data from such three DBs. In addition, some service OD flows data are stored for testing the validity of service flows models.

Once model fitting DB is obtained, the modelling and validation processes for the models explained in Chapter 3 start. The resultant models are stored in a *Models DB*. From those models, predictions can be performed for different services, OD pairs, and time instants. These predictions can be compared with previously stored testing data to finally assess the validity and accuracy of the models. The analysis of the goodness-of-fit of the models provides several statistics that are analyzed to take conclusions about the object of the simulation (e.g. evaluate the accuracy of models when some monitoring architecture is configured and service flows are assumed to be monitored at a given rate).

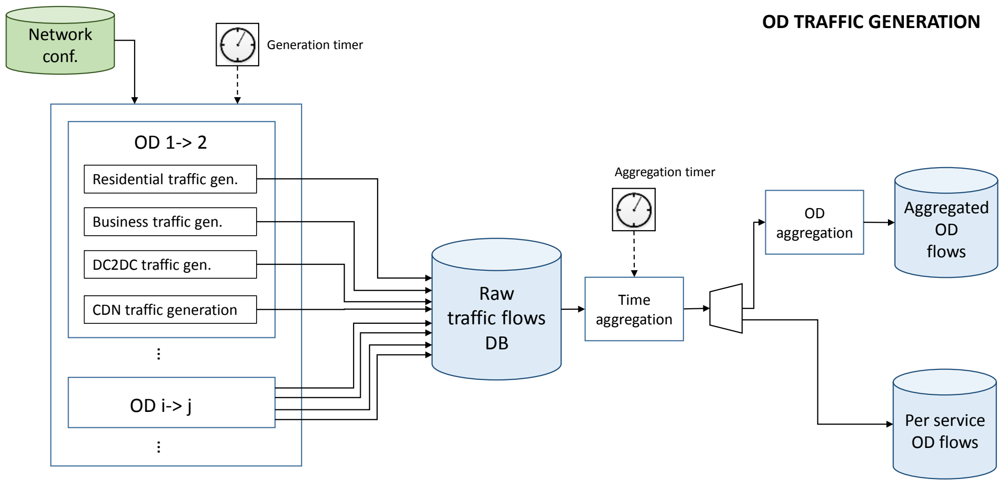


Figure 4-1: OD traffic generation of simulator.

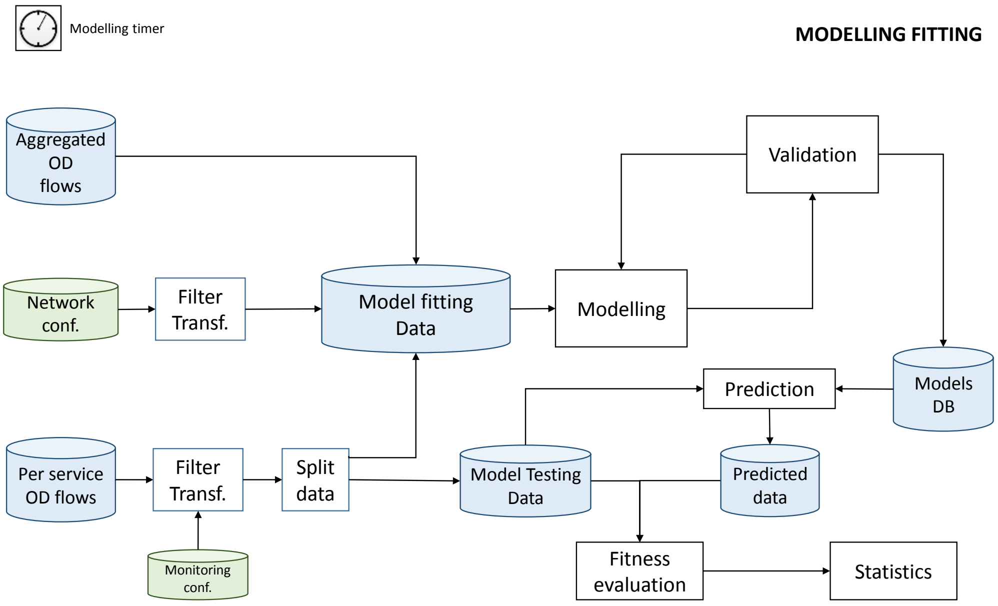


Figure 4-2: Traffic flows prediction of simulator.

4.2 Input and output details

A simulation is configured by defining the number and characteristic of locations (nodes) such as number of users for each service, the distance between the nodes, etc. In this section, all these parameters and the generated flow format are specified.

Main parameters related to the network and service profiles are:

n Positive integer. Number of nodes.

s Positive integer. Number of services.

f_s Real. The services profiles. The profile will be represented with a vector of a given length (e.g. 24) where each point represents the density of active users at a time unit (e.g. one hour).

data Positive integer. Mean of volume in GB generated by a server in data migration.

vel Positive Real. Transmission speed in Gb/s for the datacenters migration for each location.

For each of the network nodes, the simulator requires the following input parameters:

res Positive integer. Thousands of ADSL subscribers.

emp Positive integer. Thousands of employers.

cdn Positive integer. Thousands of CDN subscribers.

dc Positive integer. Number of servers in the datacenters.

cache Binary, equal to 1 if there is a cache; 0 otherwise.

gateway Binary, equal to 1 if it is Internet node; 0 otherwise.

(x,y) Real. Coordinates of the nodes in the network.

Other aspects to configure regarding the overall simulation are:

num_inspecc Positive integer. Number of DPI instances deployed in the network. Without loss of generality, we assume that one instance is able to inspect 100% of the total traffic injected in a node. However, in case of configuring less instances than nodes, inspected traffic is equally distributed in the network (e.g. if 5 DPI instances are configured in a 10-node network, then 50% of traffic at every node is analyzed)

time_inspecc Positive integer. Interval of time in time units for summarizing traffic analyzed by DPI instances.

step Positive integer. Time step in time units to generate traffic flows.

max_step Positive integer. Total time in time units for the simulation.

Given the aforementioned input parameters, the simulator generates the following output:

G numerical 2-D matrix with the aggregated flow between each pair of OD for each time step. Element $G((t\text{-}1)\cdot n + i,j)$ corresponds to the aggregated flow from node *i* to node *j* in time step *t*.

U numerical 3-D matrix with flow of all services between all OD pairs for each time step. Element $U((t\text{-}1)\cdot n{+}i,j,s)$ corresponds to the traffic flow of service *s* from node *i* to node *j* in time step *t*.

D numerical 3-D matrix with means of all services flow between all OD pairs during dpi timer. Element $D((t'\text{-}1)\cdot n + i,j,s)$ corresponds to the mean traffic flow of service *s* from node *i* to node *j* in inspection time step *t'*.

models structures with a set of candidate models for each services.

U' 3-D numerical matrix with the predict value by the models. Element $U'((t\text{-}1)\cdot n + i,j,s)$ corresponds to the predicted traffic flow of service *s* from node *i* to node *j* in inspection time step *t*.

e numerical vector with the prediction error. *e*(i) correspond to the error of day *i*.

4.3 Traffic Generation

When the simulator receives the input parameters, it generates service traffic flows in the network with the following patterns:

Residential traffic between two nodes is generated following the gravity model exposed in 3.2.3, the mass terms are the population of the adsl subscribers in both nodes: *resi* and res_j; and the d_{ij} is the Euclidian distance between the nodes.

Business traffic between two nodes is generated also with the gravity model, but the mass terms are the number of employers in both nodes: emp_i and emp_j; and the d_{ij} is the Euclidian distance between the nodes as the previous case.

CDN traffic between two nodes is generated following the weighted model exposed in 3.2.2, where the weight is the number of cdn subscribers in the destination node (cdn_j).

The *DC2DC* traffic is the most complex one. First, consider that the datacenters have two sizes: large datacenter, which have more than or equal to 10000 servers, and small datacenter, which have a range of 1000 to 9999 servers. Then, the datacenters have a regular traffic following multiplicative weighted model taking the number of server in datacenters in both nodes as the weights: *dci* and dc_j.

Additionally, it has extra traffic flow when a data migration occurs. The data volume for migration is proportional to the *DC2DC* profile and number of servers in the origin location (*i*):

$$g_D(i,j,t,x) = dc_i \cdot data_i \cdot f_s(t) \quad , \tag{4.1}$$

where *data*$_i$ denotes the mean of data volume generated by a server in the origin node. Then, it generates an additional traffic equal to the transmission speed until the data is migrated:

$$g_D(i,j,t,time_i,vel_i,x) = vel_i \quad t \le time_i \quad , \tag{4.2}$$

with *vel*$_i$ denotes the transmission speed for the location *i* and *time*$_i$ the time step when the data migration of location *i* will terminate. Note that if a migration from location *i* to location *j* occurs, then the extra traffic of migration from location *i* to the other destination (different than *j*) will be 0 since the datacenter can only have migration with one datacenter at a time point. So the function *g(i,j,t,x)* also depends on *g(i,j,t-1,time).*

For the simulator, if a data migration occurs or not is a probabilistic event, the probability that a migration from a large datacenter (p_i = *0.5*) is higher than migration from a small one (p_i = *0.3*).

$$g_D(i,j,t,g_{t-1},vel_i,time,p_i) = \begin{cases} g_{t-1} & t \le time_i \\ 0 & t > time_i \quad P(p > p_i) \\ vel_i & t > time_i \quad P(p \le p_i) \end{cases} \quad , \tag{4.3}$$

And there will be only one data migration with larger volume at night (8:00 p.m. to 6:00 a.m.) for each datacenter.

$$g_N(i,j,t,g_{t-1},vel_i,time,p_i,mig_i) = \begin{cases} 0 & mig_i = 1 \\ g_{t-1} & mig_i = 0 \quad t \le time_i \\ vel_i & mig_i = 0 \quad p \le p_i \end{cases} \quad , \tag{4.4}$$

where *mig*$_i$ denotes if a migration occurred at night (it takes value 1 once the night migration terminates).

Then, the extra flow for *DC2DC* traffic can modeled with

$$g(i,j,t,g_{t-1},vel_i,time,p_i,mig_i) = \begin{cases} g_D & t \in D \\ g_N & t \notin D \end{cases} \quad , \tag{4.5}$$

where *D* denotes the interval of daytime hours (from 6:00 a.m. to 8:00 p.m.).

Finally, the formulation for *DC2DC* service traffic is an evolutionary model as explained in 3.2.4 with *g* as an additive function:

$$Y_{ijs}(t) = dc_1 \cdot dc_2 (D_t + N(0, \frac{D_t}{100})) + g \quad , \tag{4.6}$$

For the traffic of a service generated by a node, it has a traffic flow of the same service from the location to Internet and it also receive traffic from Internet. These two flows are proportional, in different proportion, to the total traffic of the respective service generated by the node. The formulation for the traffic received from Internet follows the dependent model explained in 3.2.5, setting the Internet as node *p*:

$$Y_{pis}(t) = \sum_{\substack{ij \in OD_s \\ j \neq p}} c_{in}(s) \cdot Y_{ijs}(t) \quad . \tag{4.7}$$

And for traffic goes to Internet:

$$Y_{ips}(t) = \sum_{\substack{ij \in OD_s \\ j \neq p}} c_{out}(s) \cdot Y_{ijs}(t) \quad . \tag{4.8}$$

The $c_{in}(s)$ and $c_{out}(s)$ are two scalars independent to OD pair *ij*, it only depends on *s*; and $Y_s(i,j',t)$ denotes the traffic flow of service *s* generated by OD pair (*i,j*).

For simplicity, we assume that the service profiles do not evolves during the simulation.

The services traffic in our scenarios has the following connectivity:

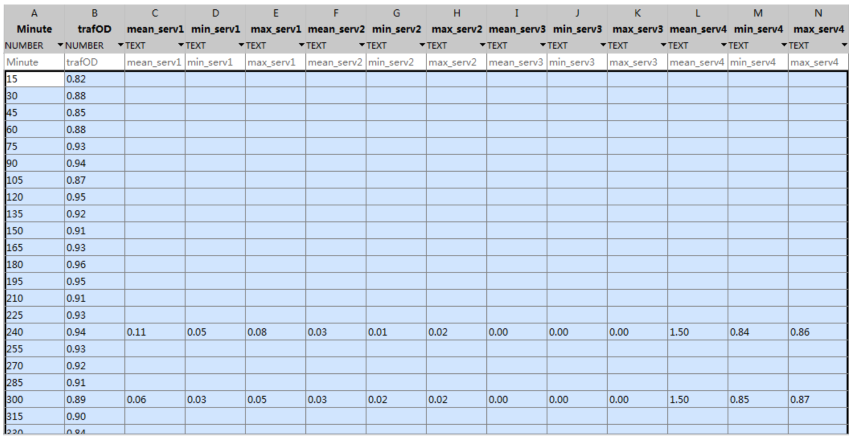
- All the nodes with *res > 0* interchange *residential* traffic among themselves.
- All the nodes with *emp > 0* interchange *Business* traffic among themselves.
- There is only *CDN* traffic from nodes with a cache to other nodes with *cdn > 0*. But there is not *CDN* traffic between network nodes and the Internet node.
- For the nodes with *dc > 0*:
    - There are regular *DC2DC* traffic among themselves.
    - There are data migration among datacenters with same size.
    - There are data migration from small to large datacenter, but not in the other way.

4.4 Modelling

On the one hand, the OD traffic flows is stored in the database at each aggregation timer, e.g., every 15 minutes. On the other hand, there are a fixed number of DPI instances extracting service flow information at the nodes (random selected)at another rate (e.g. summaries every hour) . Consequently, the data generated has not information of the service traffic flows at equally spaced time intervals, because some OD pair presents missing values. The Figure 4-3 shows one such example of the generated data, where one can observe that the data provided by DPI (the means, minimums and maximums of each service) can be very sparse. Recall that each row represents one collected measure of aggregated OD traffic.

The time series analysis is seems not to be appropriate to model this data due to the sparsity as explained in 3.3.1. In contrast, linear model is a good approach since we can use all the other covariates. Thus, we can use the formulation (3.19) for our model without the values of response variable in previous time steps. Then, for modelling, we use the information extracted from DPI to construct a linear regression model and predict the service matrix for each time step in the future.

For the construction of linear models, we use all the input data mentioned in the previous section, also the aggregated flow matrix (*G*) and DPI matrix (*D*) as variables to predict the values of the service's flow matrix (*U*). As the profile of the services are periodic with period of 1 day, then we can take the relative hour on the day as an explanatory variable. Applying Algorithm 3-3, some power terms of these variables can be also introduced as explanatories.



A	B	C	D	E	F	G	H	I	J	K	L	M	N
Minute	**trafOD**	**mean_serv1**	**min_serv1**	**max_serv1**	**mean_serv2**	**min_serv2**	**max_serv2**	**mean_serv3**	**min_serv3**	**max_serv3**	**mean_serv4**	**min_serv4**	**max_serv4**
NUMBER	NUMBER	TEXT	TEXT	TEXT	TEXT	TEXT	TEXT	TEXT	TEXT	TEXT	TEXT	TEXT	TEXT
Minute	trafOD	mean_serv1	min_serv1	max_serv1	mean_serv2	min_serv2	max_serv2	mean_serv3	min_serv3	max_serv3	mean_serv4	min_serv4	max_serv4
15	0.82												
30	0.88												
45	0.85												
60	0.88												
75	0.93												
90	0.94												
105	0.87												
120	0.95												
135	0.92												
150	0.91												
165	0.93												
180	0.96												
195	0.95												
210	0.91												
225	0.93												
240	0.94	0.11	0.05	0.08	0.03	0.01	0.02	0.00	0.00	0.00	1.50	0.84	0.86
255	0.93												
270	0.92												
285	0.91												
300	0.89	0.06	0.03	0.05	0.03	0.02	0.02	0.00	0.00	0.00	1.50	0.85	0.87
315	0.90												
330	0.84												

Figure 4-3: Illustration of generated data of an OD pair.

4.5 Validation

For the validation of the models, it applies the stepwise regression procedure as explained in Algorithm 3-1, with AIC criteria to select the set of explanatory variables and testing up to cross-product terms of these variables. That is, it searches the set of variable from all the explanatory variables and their cross-product terms, which has the model with minimum AIC information. It also applies

Box-Cox transformations to the response variable to make it a normal-like distributed data.

The simulator applies Algorithm 3-1 to get a set of candidate models. To evaluate the goodness of the models, it compares the predict matrix for services flow *U'* and the real flow matrix *U* and computes the mean error for each services with the formula exposed in (3.35).

We consider two errors: the absolute error, using (3.35), and the relative error (the absolute error divided by the real value of the flow); we will use the absolute one to compare models for different approach and the relative one to evaluate the prediction error for different services.

4.6 Summary

In this chapter, the details of the main blocks and DBs (and the relationship among them) included in our designed simulator have been firstly presented. The inputs and outputs of the simulator were presented defined. Choosing linear regression as most appropriate modelling tool, stepwise linear regression with AIC criteria for model selection has been implemented. At last, the process to evaluate the goodness of the candidate models is exposed.

In the following chapter, the utility of the simulator will be evaluated through numerical results over an illustrative case study.

Chapter 5.

Numerical results

This chapter presents a case study where different monitoring and data analytics configurations are evaluated from the perspective of service flow modelling. By means of numerical results obtained with the simulator, the accuracy of models and the amount of monitoring data needed to obtain a target goodness-of-fit threshold are evaluated for different configurations of the service flow monitoring function. Finally, the sensitivity of some key explanatory variables in the service traffic models is analyzed.

5.1 Case study

As we introduced in Chapter 2, there are two main (and opposite) data monitoring architectures, i.e. *Centralized* and *distributed*. With those concepts in mind, in this study we compare three different approaches for collecting monitoring data and obtaining predictive traffic models, namely *Network*, *Node* and *OD*. The main idea behind each option is following explained:

- The *Network* approach takes all the available information to construct a model. Thus, each of the services is characterized by a unique model where explanatory variables include data from the source and destination locations of the OD that wants to be predicted. Note that this model requires centralizing monitoring data to a common repository, where also locations data need to be synchronized to allow model fitting and predictions.
- The *Node* approach assumes knowing available partial information. Basically, it requires from traffic monitoring data at sources and some

destination location data. This approach can be deployed either in a centralized way or in a distributed one. If the latter is chosen, although some location data needs to be spread and replicated at every location, traffic monitoring data collected locally can remain in local repositories.

- The *OD* approach consider obtaining models just from local information at nodes. This is the best candidate to implement a fully distributed scheme where data is stored locally and models per each OD are obtained only by means of monitoring traffic data at source locations.

Table 5-1 shows the amount of models that need to be fitted under every approach, where $|S|$ represent the amount of service and $|N|$ the amount of network locations. Note that the simpler the model is (in terms of considered explanatory variables) the higher is the amount models to fit. For the sake of a fair comparison, we assume that models under the Node approach can use a subset of the explanatory variables of the Network model and similarly, OD models use a subset of the Node models one (as illustrated in Table 5-2).

Table 5-1: Number of models for each of the approaches

Approach	**Number of models**
Network	$\lvert S\rvert$
Node	$\lvert N\rvert \cdot \lvert S\rvert$
OD	$\lvert N\rvert \cdot (\lvert N\rvert -1) \cdot \lvert S\rvert$

Table 5-2: Explanatory variables for each of the approaches.

	Hour	***TrafOD***	***Destination node information***				***Origin node information***			
			***res**$_j$*	**…**	***dist**$_j$*	**gravity**	***res**$_i$*	**…**	**…**	***dist**$_{ij}$*
OD	X	X								
NODE	X	X	X	X	X	X				
NETWORK	X	X	X	X	X	X	X	X	X	X

To represent different configurations of de-aggregated service traffic monitoring, we configure different filters to allow considering a pre-defined proportion of service monitoring traffic for model fitting. In this way, we emulate different deployments of DPI VNF instances into computing network resources. Without loss of generality, we assume that every considered approach has its best DPI VNF deployment, e.g. *OD* models can take advantage of deploying few resources in local computing nodes, whereas a *Network* approach could concentrate both DPI and traffic modelling resources in centralized and powerful DCs.

5.2 Reference scenario

Figure 5-1 shows the reference network selected for evaluation of the case study. It contains 6 node locations with distinct characteristics: 3 of them are collocated with a DC and 1 is collocated with a large cache node. Moreover, node 6 interconnects the network with the Internet.

For this scenario, we considered that each user generates around of dozens of Kb/s for *Residential* and *Business* services; and Mb/s for *CDN* service. We establish dozens of GBs as the volume generated by a datacenter's server in a migration on the day and hundreds of GBs at the night. The data volume is transmitted with speed 5 Gb/s on the day and 20 Gb/s at the night for small datacenters; the transmission is double quickly at the night for large datacenters. We assume that the service traffic flows coming from the Internet are equivalents to the total service traffic generated by the node ($C_{in}(s) = 1$), except *CDN* since we configured it has no interchange flow with Internet. But the service traffic go to the Internet has different proportions: 0.6 for *Residential*, 0.4 for *Business* and 0.2 for *DC2DC*.

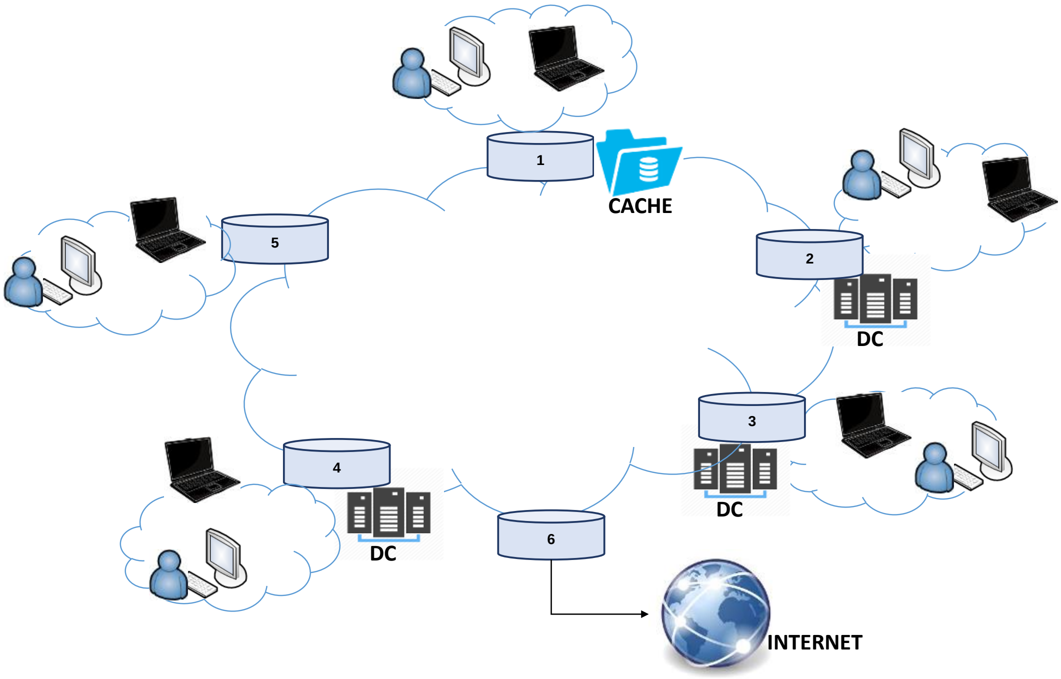


Figure 5-1: The reference scenario

Table 5-3 shows the number of users for each service in every node (in thousands for users of *Residential*, *Business* and *CDN*). We considered that Internet node has the service user numbers equal to the total users of the respective service in the network. With this configuration, where every node has different proportion of each service users, we are able to generate a wide variety of aggregated and de-aggregated traffic flows for OD pairs. In fact, although that the scenario seems to have few node locations, it generates 4 service flows for 30 distinct OD pairs, that is, 120 service flows altogether.

	NODE 1	***NODE 2***	***NODE 3***	***NODE 4***	***NODE 5***
res	*80 000*	*10 000*	*50 000*	*100 000*	*250 000*
emp	*140 000*	*100 000*	*300 000*	*60 000*	*160 000*
cdn	*40 000*	*80 000*	*90 000*	*110 000*	*200 000*
Dc	*0*	*1 000*	*1 000*	*10 000*	*0*

Table 5-3: Users of each services for the nodes in the network.

We generated simulations of 15 days, generating traffic flows every minute and with a time interval for monitoring traffic aggregation equal to 15 minutes. In terms of data to manage for modelling, one can easily observe that each of the distinct OD flows has 96 records for a day. Since simulations last for15 days, 1440 records for each service flow for different OD pair are generated. Then, modelling procedure must manage data sets containing 1440·120 = 172800 records, which is a considerable volume of data.

An example of generated service traffic flows are shown in Figure 5-2. The traffic volume generated by each service is between 20-30% of the total traffic. The 23% of the aggregated traffic volume corresponds to *Residential* service, 23% to the *Business*, 24% to the *CDN* and 30% to the *DC2DC* service. Nonetheless, according to the connectivity, there is only *CDN* traffic between node 1 and the others nodes; and the *DC2DC* service only occurs between DC nodes. So the traffic generated by a pair of OD has much higher volume in *CDN* and *DC2DC* than *Business* and *Residential.*

Next sections will be devoted to analyze that amount of generated traffic data and take conclusions about the proposed data pre-processing and modelling procedures, and evaluate the strong and weak aspects of each proposed data analytics approach.

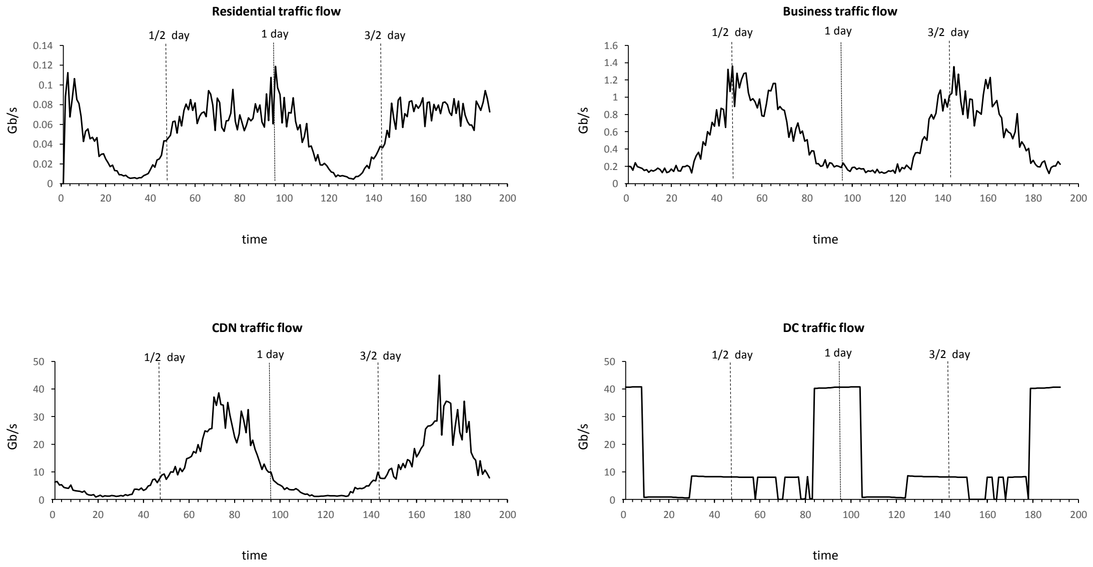


Figure 5-2: Example of generated service flows (2 days).

5.3 Data preprocessing and transformation

To evaluate whether using different power terms of the variables or not using, we processed scatterplots for explanatory and response variables to see the relationship between them. Some of the resultant scatterplots are shown in Figure 5-3, where we observe that the most of the explanatory ones only take few values and no relation is concluded. Apart from this observation, the R-squared value did not significantly improve when new terms of these variables are added. Figure 5-4 shows the R-squared evolution of models for *Business* service traffic when new terms of *emp* are added. In view of the figure, we can conclude that it is not necessary to apply such data transformation algorithm to this explanatory variable. Besides this, we observed that there are some relationship between the explanatory variables, such as *res* and *cdn* that have a strong linear relation. This fact causes to linear models that include one or the other will not offer difference and we will take it account in the next section for methodology evaluation.

After analyzing all the variables, we consider that the only two interesting variables to add significant power terms are *Hour* and *trafOD*. Figure 5-5 shows the scatterplots of these variables with *Business* as example; in this figure, the *Hour* seems to have quadratic correlation with *Business*; the *trafOD* seems linear with *Business* at beginning but the linearity fails when *trafOD* is large. Then, we applied transformation to these two variables and set $\delta = 0.05$.

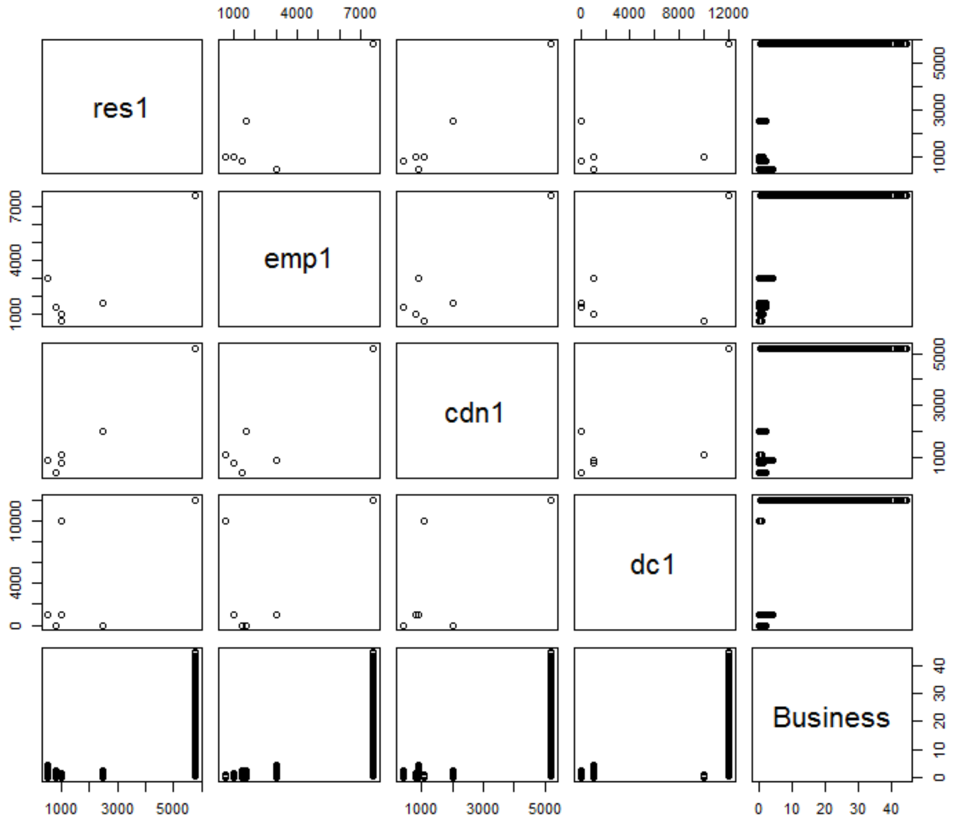


Figure 5-3: Scatterplots of different variables w.r.t *Business*

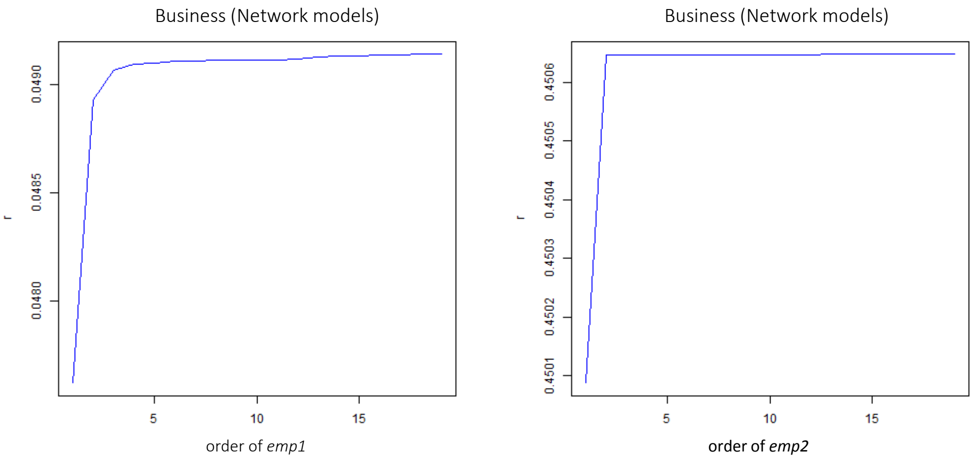


Figure 5-4: R-squared plot of *Business* with *emp1* and *emp2*.

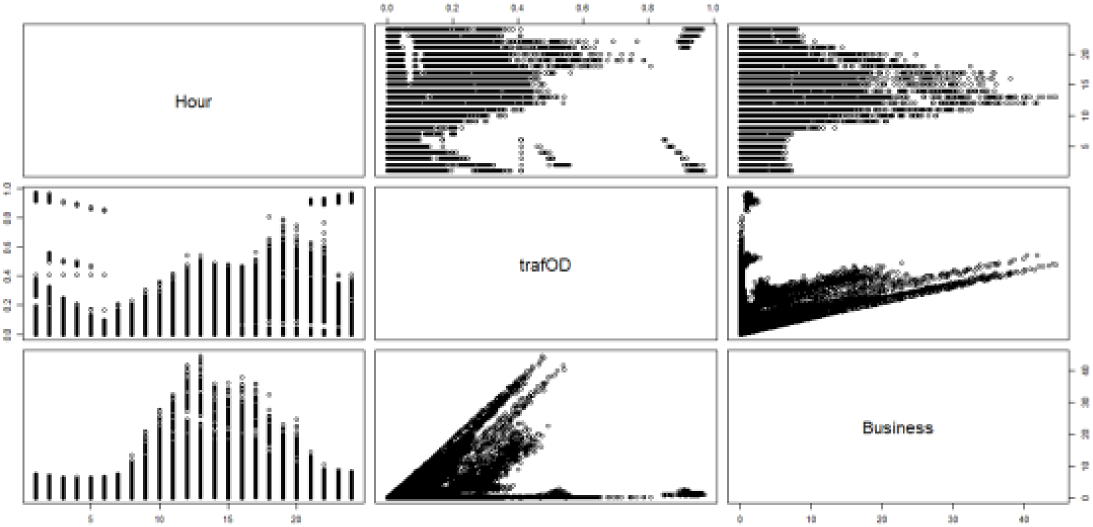


Figure 5-5: Scatterplots between *hour* and *trafOD* and *Business*.

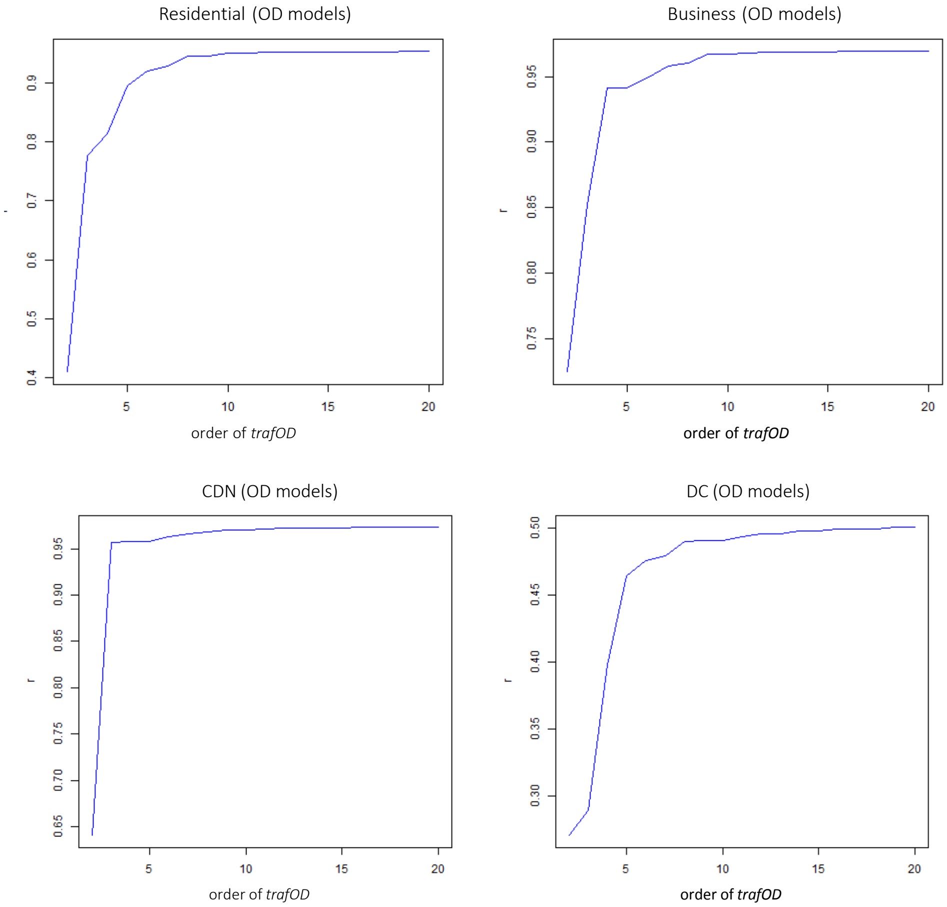


Figure 5-6: R-squared plot for OD models.

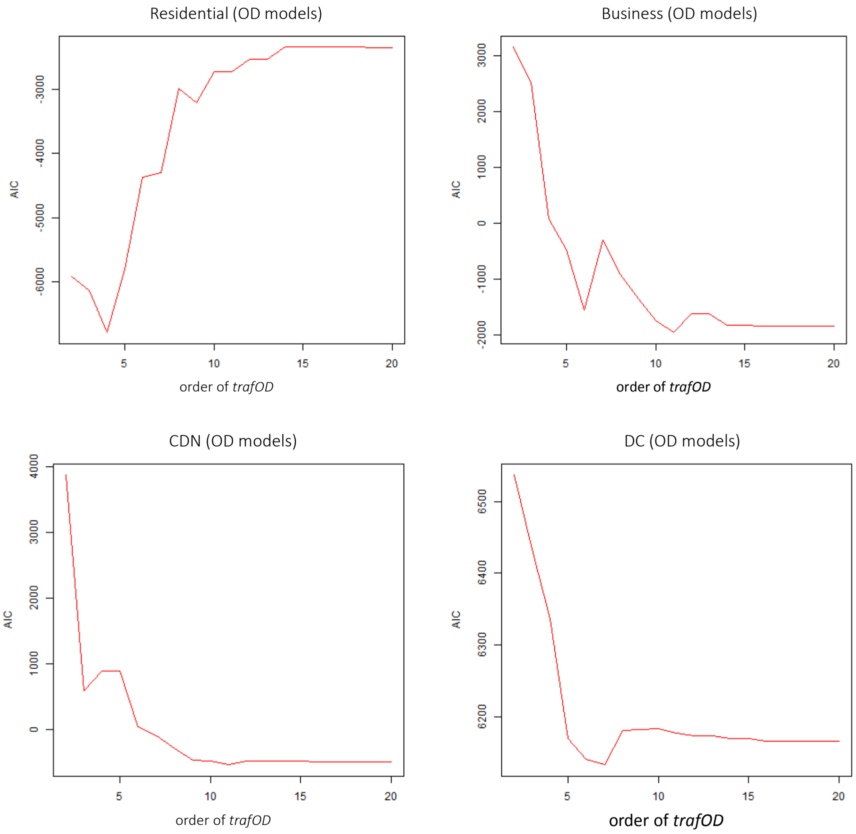


Figure 5-7: AIC plot for OD models.

Figure 5-6 shows the evolution of R-squared values when new terms of *Hour* are added for OD models. As we can observed, the R-squared has a great improvement when new terms are added. The Figure 5-7 shows the evolution of AIC information, for *Residential* service, it has a minimum at order 4; for the cases of *Business* and *CDN,* at order 11; and *DC2DC* has it at order 7. The algorithm concluded to add hour up to order 8 for *Residential*; up to order 11 for *Business* and *CDN*; and 7 for *DC2DC*. But to be fair in the modelling of each service, we add the highest term (order 11) to all the models. There may be too much variables for *Residential* and *DC2DC*, but the stepwise process will take over those variables that are the most appropriate ones to construct the model. We do the same analysis for the variable *trafOD* and concluded to add terms up to order 15.

We also combine some of the predictive variables to get new ones. For example, the distance with number of users of a service in such way to get a gravity model. Although we know that the simulator generates some services with gravity model, and we are supposed to be without this information in a real case, we can do it because there is gravity model theory behind this. The matrix of scatterplots for gravity models is shown in the Figure 5-8 we see that the gravity model has more or less a linear correlation with *Residential* and *Business* services, what is really helpful introducing this variable to the linear model. But it seems not to have a linear correlation with CDN and DC2DC services. Seeing that, we decide to use gravity model as predictive variable in models for *Residential* and *Business*.

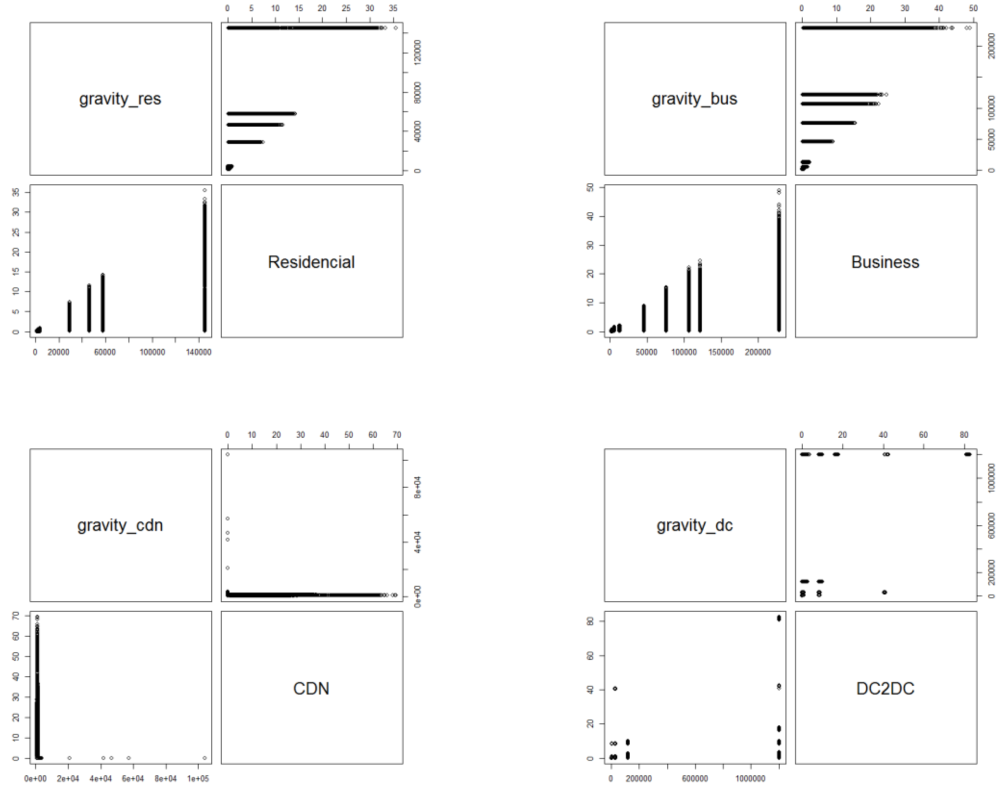


Figure 5-8: Scatterplots between gravity models and the services.

5.4 Model fitting methodology evaluation

This section is devoted to evaluate the performance of the stepwise regression procedure. Since the methodology is the same for all the considered approaches, the evaluation is done with an exhaustive analysis of the process for each service model in the *Network approach*, which includes all the explanatory variables. Outcomes and conclusions of this section will be applied in further sections to compare different approaches.

After applying the data transformation process explained before, 39 explanatory variables are initially considered; that is, a total of 1482 cross-product terms that stepwise regression can choose. For the sake of simplicity, we decided to avoid variable interactions and consider only linear relations among variables. Next subsections present results for each of the considered services.

5.4.1 Residential

The result of the stepwise process with linear variables for the *Residential* service is shown in Table 5-4. We see that the predictive variables considered are: *res, emp, cdn, dc, dist, gravity, gateway, cache, hour and trafOD*. The variables *hour, trafOD, dist, gravity* and *res* are clearly related variables. In contrast, other variables such as *emp*, *cdn* or *dc* seem not related with *Residential* traffic flow. This facts due to the relationship among these variables with *res* as explained in the previous section. Another fact that we observed is that the prediction of *Residential* flow depends on *trafOD*, which depends on other variables, so his values can be modified by other values. For example, a large value of *emp* in both nodes generate a larger *Business* traffic flow, and it contributes a larger *trafOD* flow, what makes a larger prediction value of *Residential* traffic flow. The same thing happens with *gateway* and *cache* since *trafOD* flows that goes to Internet or that have *CDN* are much higher than traffic without these conditions.

Table 5-4: Stepwise process for Residential service

```
1. Adding dist                       AIC = 7713.141
2. Adding res1                       AIC = 6772.837
3. Adding Hour2                      AIC = 6280.2316
4. Adding res2                       AIC = 6038.9278
5. Adding Hour                       AIC = 5782.4047
6. Adding Hour3                      AIC = 4406.2143
7. Adding gateway                    AIC = 4011.9741
8. Adding Hour8                      AIC = 3797.7978
9. Adding Hour4                      AIC = 3134.9886
10. Adding Hour7                     AIC = 2095.1988
11. Adding cache                     AIC = 1762.9129
12. Adding dc1                       AIC = 1362.0558
13. Adding emp2                      AIC = 1166.5572
14. Adding cdn1                      AIC = 825.9848
15. Adding gravity_mult_res          AIC = 779.839
16. Adding gravity_mult_bus          AIC = 756.6801
17. Adding Hour5                     AIC = 738.4066
18. Adding trafOD6                   AIC = 736.8389
19. Adding trafOD5                   AIC = 733.9518
```

Interestingly, we observe that there are correlations between service traffic even when we did not make it deliberately in the generation. This fact makes this analysis more complex since the real correlations between the variables are unknown.

Details of the final model are below:

```
Generalized Linear regression model:
    Residential ~ [Linear formula with 20 terms in 19 predictors]
    Distribution = Normal

Estimated Coefficients:
                         Estimate         SE         tStat       pValue
                        ___________   __________   _______   ___________
    (Intercept)              -2.0569       0.050523     -40.713    3.0599e-286
    Hour                     0.56884       0.042173      13.488      3.042e-40
    Hour2                   -0.40939        0.02151     -19.032     3.8618e-76
    Hour3                   0.048641      0.0044834      10.849     6.6708e-27
    Hour4                -0.00079789     0.00043404     -1.8383       0.066122
    Hour5                -9.3828e-05     1.7744e-05     -5.2878     1.3314e-07
    Hour7                 2.8425e-07     2.0201e-08      14.071     1.5694e-43
    Hour8                -6.7549e-09     4.0776e-10     -16.566     6.2852e-59
    dist                  -0.0026907     3.0501e-05     -88.217              0
    gravity_mult_res     -1.9934e-06     2.8734e-07     -6.9374     4.9185e-12
    gravity_mult_bus       9.618e-07     1.8071e-07      5.3225      1.103e-07
    gateway                  -1.0833       0.053448     -20.268     1.8902e-85
    cache                  -0.783309       0.025197      3.3064     0.00095687
    res1                   0.0016892     3.0707e-05       55.01              0
    cdn1                  -0.0010033     4.4252e-05     -22.672    8.6436e-105
    dc1                   5.4624e-05     1.8074e-06      30.223    2.2651e-174
    res2                  0.00066426     9.7371e-06       68.22              0
    emp2                 -0.00018843     6.9808e-06     -26.992     3.844e-143
    trafOD5               4.4488e-10     2.2464e-10      1.9804       0.047755
    trafOD6              -5.0328e-12      2.424e-12     -2.0762        0.03796

2880 observations, 2860 error degrees of freedom
Estimated Dispersion: 0.0821
F-statistic vs. constant model: 9.53e+03, p-value = 0
R-squared: 0.9845
```

As we can see, the variables *trafOD* and *hour* have both positive and negative coefficients in terms of different orders in order to approximate the values *Residential* with polynomial terms; the *res* have positive coefficients because it contribute positively in the model (the higher the amount of users the higher the traffic is); *dist* has a negative contribution in the predictive model because it is inverse proportional to the service flow according to the gravity model; the variables *cache*, *gateway*, *emp*, *cdn* and *dc* have negative or small coefficients in order to correct the linear dependence between *trafOD* and *Residential* as explained before; at last, gravity model for residential has a negative coefficient and business gravity model has positive coefficient, what is strange since the service flow is generated with residential gravity model. It is probably due to the complex correlations between the variables: the relationship among *res* and *emp* as explained in the previous section, the dependence of *trafOD*, etc. Thus, all the variables of this model seems to be significant with relevant coefficients.

The stepwise process and the resulting linear model with cross-terms are available in Apendix A. The AIC began at *7679* and it decreased to -1213 when process finished. It selected 69 variables in total of 1482 variables, and the R-squared worth 0.9940, that is, the process has selected less than 5% of the variables, explaining more than 99% of the response variable variation, what is another proof that the method works properly. The variables that involve in the model are the same as the linear case with interactions between them.

For the residuals tests in this model, the Jarque-Beras test rejects the normality of the residual distribution, but we can see in Figure 5-9a, which shows the normal plot of the residuals, that it is not too far from a normal distribution. The *Ljung-Box Q-test* accept the null hypothesis and confirm the residuals of this model are independent, as shown in Figure 5-9b, there is not strong autocorrelation in these residuals.

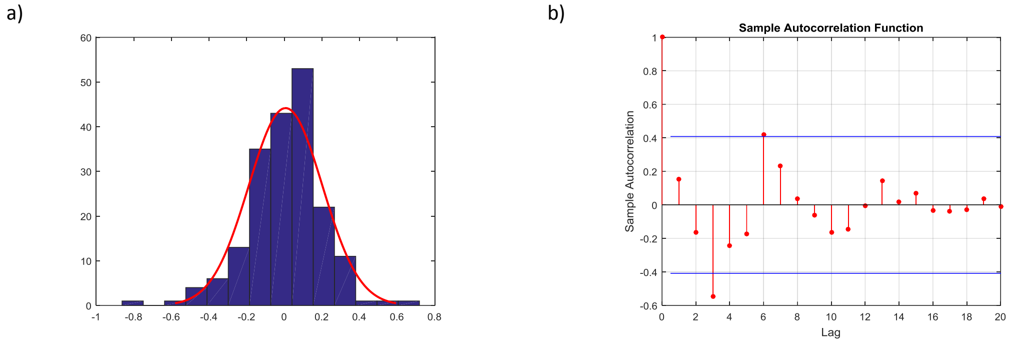


Figure 5-9: Normal plot and autocorrelation for residuals of *Residential* service.

Although the normality test fails, we did not reject directly the validation of our methodology, and we analyzed the prediction for different OD pairs to evaluate the goodness-of-fit of the model. Figure 5-10 illustrates a representative example of the *Residential* traffic flow prediction with this model. In this figure, one can observe that the model fits well in general, although it sometimes underestimates traffic and makes some bad prediction in the highest traffic flow points, particularly, at beginning of a day.

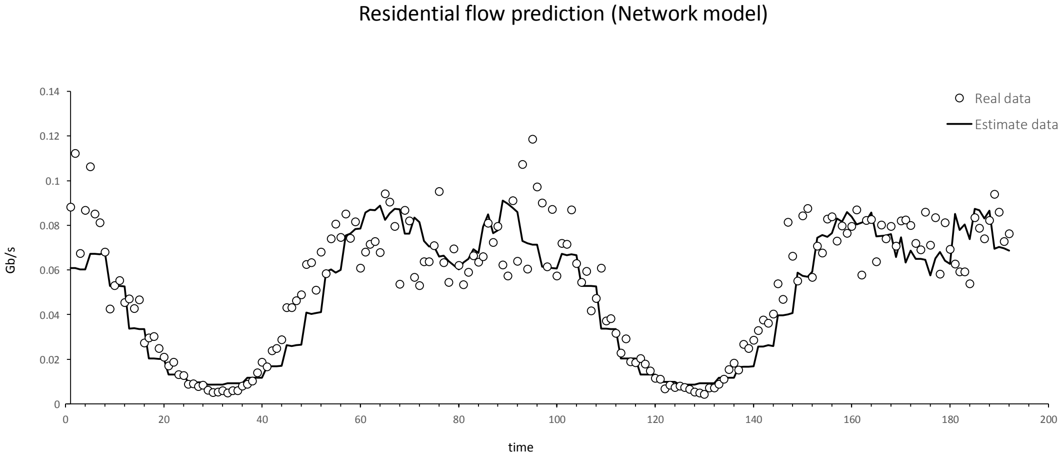


Figure 5-10: *Residential* flow prediction (from node 1 to node 5).

5.4.2 *Business*

The result of the stepwise process with linear variables for *Business* service is shown in Table 5-5. We see that the predictive variables considered are: *res, emp, cdn, dc, dist, gravity, gateway, cache, hour and trafOD*. The explanation for these variables is similar to the *Residential* service, but now the mass term is *emp* instead of *res*. The final model is detailed below:

```
Generalized Linear regression model:
    Business ~ [Linear formula with 25 terms in 24 predictors]
    Distribution = Normal

Estimated Coefficients:
                         Estimate          SE          tStat       pValue   
                        ___________    __________    _______    ___________

    (Intercept)             -4.3456      0.043903    -98.982              0
    Hour                    0.40272      0.036134     11.145     2.8697e-28
    Hour2                  -0.40942      0.019505     -20.99      4.346e-91
    Hour3                   0.11264      0.004388      25.67    6.1769e-131
    Hour4                 -0.012697    0.00048214    -26.336    5.0681e-137
    Hour5                0.00067658    2.6466e-05     25.564    5.6417e-130
    Hour6               -1.5109e-05    6.1728e-07    -24.477     2.859e-120
    Hour8                3.4764e-09    1.5435e-10     22.523    1.5619e-103
    dist                 -0.0023477     2.657e-05    -88.362              0
    gravity_mult_res     2.4078e-06    2.3877e-07     10.085     1.5943e-23
    gravity_mult_bus    -3.0784e-06    1.5097e-07    -20.391     2.1474e-86
    gateway                 -2.3787      0.067121    -50.337              0
    cache                   -0.9937      0.038773      51.42              0
    res1                 -0.0018646    6.4943e-05    -28.711    1.7464e-159
    emp1                  0.0005121    1.8165e-05     28.192    1.7737e-154
    cdn1                  0.0029533    9.5772e-05     30.837    1.7598e-180
    dc1                 -5.2702e-05    3.3739e-06     -15.62     7.6481e-53
    emp2                  0.0007266    7.7372e-06     93.911              0
    cdn2                -0.00010091     1.141e-05    -8.8445     1.5774e-18
    dc2                 -3.8471e-05    1.7038e-06    -22.579    5.2997e-104
    trafOD                 0.034956     0.0044648     7.8294     6.8544e-15
    trafOD2               -0.002839    0.00038578    -7.3593     2.4052e-13
    trafOD3              6.8313e-05     1.193e-05     5.7261     1.1344e-08
    trafOD4             -5.4038e-07    1.2663e-07    -4.2676     2.0404e-05
    trafOD6              1.0892e-11    4.4684e-12     2.4375       0.014851

2880 observations, 2855 error degrees of freedom
Estimated Dispersion: 0.0539
F-statistic vs. constant model: 7.76e+03, p-value = 0
R-squared: 0.9849
```

The stepwise process and the resulting linear model with interaction are available in Apendix A. The AIC of this model began at 6982 and decreased to -2400. The model selected 55 variables and R-squared worth 0.9941.

For the tests of residuals, the Jarque-Beras test rejects the normality of the residual distribution, however, it rejects with a p-value of 0.0377. In Figure 5-11a shows the normal plot of residuals, one can see that it closely resembles a normal distribution. The Ljung-Box Q-test accepts the null hypothesis and confirm that the residuals of this model are independent. As shows in Figure 5-11b, we see there is not strong autocorrelation in the residuals.

An example for *Business* traffic flow prediction is shown in Figure 5-12. We can see that the model has good fit in general even when it behaves slightly different in the rush time.

Table 5-5: Stepwise process for Business service.

```
1. Adding emp1                        AIC = 7235.4725
2. Adding emp2                        AIC = 6521.536
3. Adding dc2                         AIC = 6233.9602
4. Adding Hour                        AIC = 5951.5569
5. Adding Hour4                       AIC = 4564.1031
6. Adding dist                        AIC = 4178.3773
7. Adding gateway                     AIC = 3392.377
8. Adding cache                       AIC = 2895.1728
9. Adding cdn1                        AIC = 2717.327
10. Adding res1                       AIC = 2530.5759
11. Adding gravity_mult_bus           AIC = 2389.9035
12. Adding Hour2                      AIC = 2252.3477
13. Adding Hour3                      AIC = 684.1794
14. Adding dc1                        AIC = 524.8696
15. Adding res2                       AIC = 500.883
16. Adding gravity_mult_res           AIC = 451.8432
17. Adding Hour5                      AIC = 434.4108
18. Adding Hour6                      AIC = 267.5157
19. Adding Hour8                      AIC = -83.1315
20. Adding trafOD6                    AIC = -93.7572
21. Adding trafOD2                    AIC = -123.4507
22. Adding trafOD4                    AIC = -143.9359
23. Adding trafOD                     AIC = -159.5645
24. Adding trafOD3                    AIC = -179.3428
25. Adding cdn2                       AIC = -179.9752
26. Removing res2                     AIC = -181.97
```

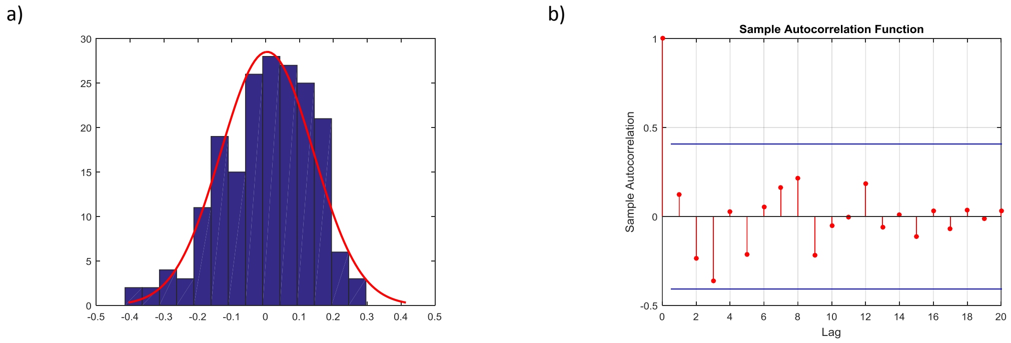


Figure 5-11: Normal plot and autocorrelation for residuals of *Business* model.

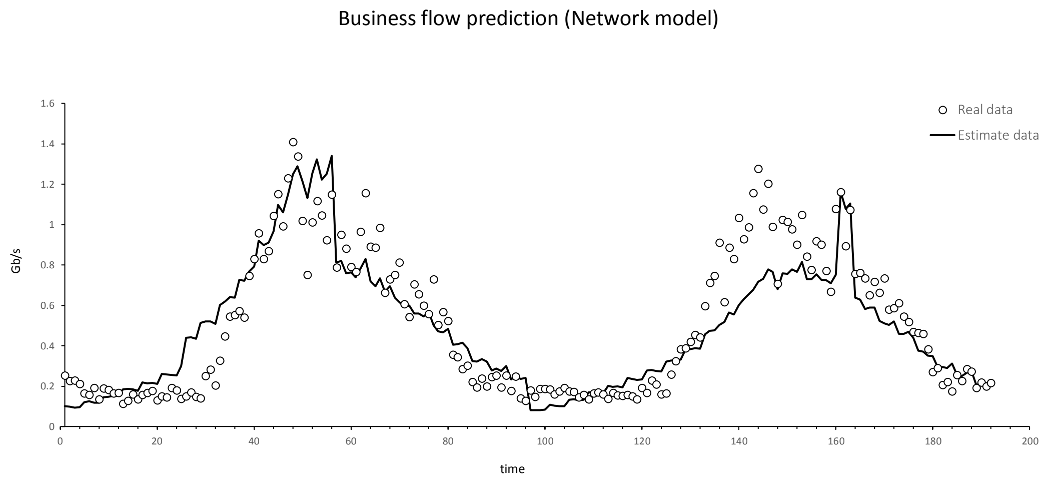


Figure 5-12: *Business* flow prediction (from node 1 to node 5)

5.4.3 CDN

The result of the stepwise process with linear variables for the *CDN* service is shown in Table 5-6. We see that the predictive variables considered are: *emp, hour and trafOD*. It is obviously the dependence on the *trafOD* since that, for a given pair of OD, the *CDN* and *DC2DC* service flows has much higher volumes than *Residential* and *Business*, as explained in 5.2. Then, the *trafOD* flow between a pair OD which has *CDN* flow is really dependent to this service flow. Details of the final model are as follows:

```
Generalized Linear regression model:
    CDN ~ 1 + Hour + Hour7 + Hour9 + Hour10 + emp2 + trafOD + trafOD5
    Distribution = Normal

Estimated Coefficients:
                    Estimate          SE          tStat       pValue
                   ___________    __________    _______    ___________

    (Intercept)              0             0          0              0
    Hour              -0.15454      0.017943    -8.6126     1.0772e-16
    Hour7          -2.5441e-08    5.4062e-09    -4.7058     3.3268e-06
    Hour9           2.0229e-10    3.3599e-11     6.0208     3.4914e-09
    Hour10         -6.6078e-12    1.0281e-12    -6.4269     3.1894e-10
    emp2           -0.00016276    2.6248e-05    -6.2007      1.228e-09
    trafOD              1.0046      0.011146     90.124    1.4208e-299
    trafOD5         -3.588e-09    9.8172e-10    -3.6548     0.00028628


480 observations, 473 error degrees of freedom
Estimated Dispersion: 1.93
F-statistic vs. constant model: 6.18e+03, p-value = 0
R-squared: 0.9874
```

Table 5-6: Stepwise process for CDN service.

```
1. Adding trafOD                    AIC = 1470.7007
2. Adding Hour7                     AIC = 1406.3492
3. Adding Hour                      AIC = 1353.2947
4. Adding Hour10                    AIC = 1322.1173
5. Adding gravity_mult_bus          AIC = 1287.8029
6. Adding Hour9                     AIC = 1270.8982
7. Adding emp2                      AIC = 1269.9552
8. Adding trafOD5                   AIC = 1269.7622
9. Removing gravity_mult_bus        AIC = 1267.8
```

One can observe that *trafOD* has a coefficient ≈*1*, what means there are a quasi linear correlation between *trafOD* and *CDN*. This is another prove of the importance of *trafOD*. The *gateway* has no effect in this service since there is no *CDN* traffic to Internet.

For the cross-product terms case, the AIC begins at 1484 and decreases to 1150. The model selected 10 variables and R-squared worth 0.9889. The stepwise process and the resulting linear model are available in Apendix A.

For the tests of residuals, the Jarque-Beras test rejects the normality of the residual distribution. In the Figure 5-13a shows the normal plot of the residuals, one can see that it has too much small residual values to be a normal distribution, what seems good since we do not want a model with much high residuals. The Ljung-Box Q-test accepts the null hypothesis and confirms that the residuals of this model are independent. As shown in Figure 5-13b, there is very small autocorrelation in these residuals.

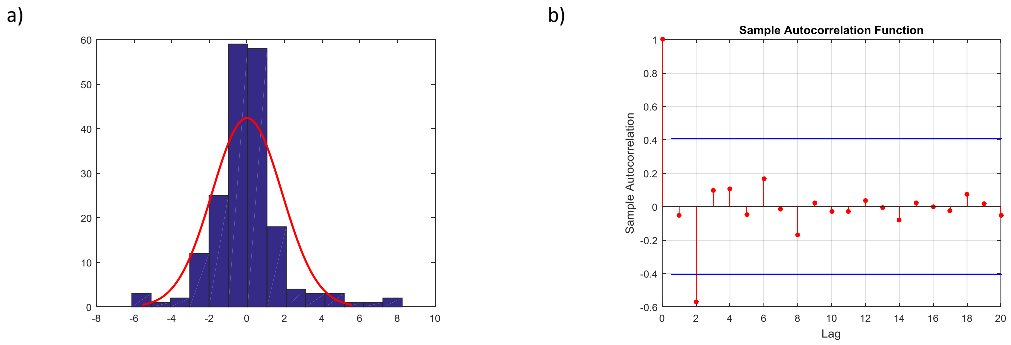


Figure 5-13: Normal plot and autocorrelation for residuals of *CDN* model.

Although the normality test of residuals fails, the prediction for this service is much fitted. An example of *CDN* traffic flow prediction is shown in Figure 5-14, where we can see that the model fits very well the traffic flow.

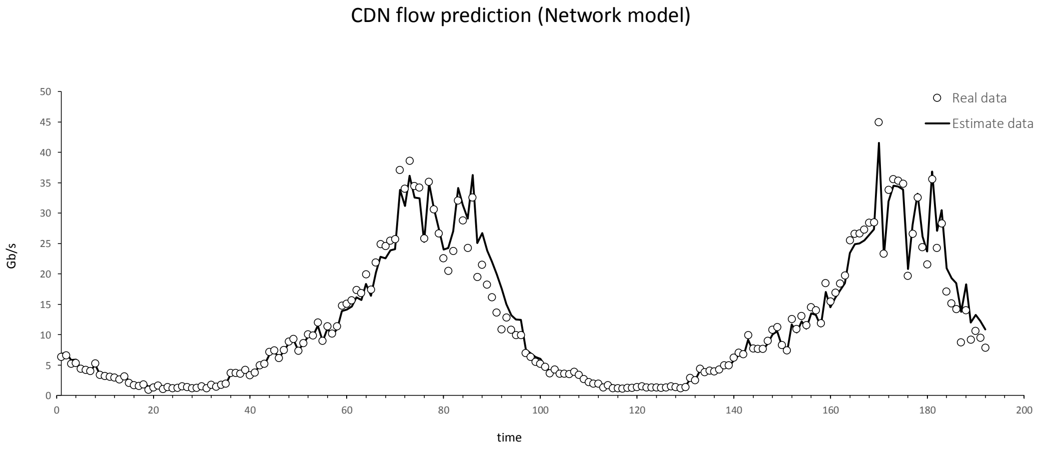


Figure 5-14: *CDN* flow prediction (from node 1 to node 5)

5.4.4 DC2DC

The result of the stepwise process with linear variables for the *DC2DC* service is shown in Figure 5-16.We see that the predictive variables considered are: *gateway, cdn, dc, gravity, hour and trafOD*. There is a strong dependence between *DC2DC* and *traOD* when a data migration occurs (in this case, the *DC2DC* traffic flow contributes the mayor part of total flow) and the regular *DC* flow is dependent on *hour*, *trafOD* and *dc*. The introduction of variable *gateway* is because the service flow to the Internet has a different generation model as previous cases. The variable *cdn* and *gravity* for business are probably introduced to correct the dependence on *trafOD* as explained before. Final model is as follows:

```
Generalized Linear regression model:
    DC ~ 1 + Hour + Hour5 + gravity_mult_bus + gateway + cdn2 + dc2 + trafOD + trafOD2 +
trafOD3 + trafOD4
    Distribution = Normal

Estimated Coefficients:
                         Estimate          SE          tStat       pValue
                        ___________    __________    ________    __________

    (Intercept)             0.89081       0.37959      2.3468      0.019108
    Hour                   -0.51093      0.032664     -15.642      4.083e-50
    Hour5                1.7669e-06    1.2934e-07      13.661    1.8282e-39
    gravity_mult_bus    -1.7623e-05    2.4242e-06     -7.2695    6.6798e-13
    gateway                 -7.1981       0.56811      -12.67    1.6166e-34
    cdn2                 0.00083792    0.00017024       4.922    9.8237e-07
    dc2                  0.00046779    4.5292e-05      10.328    5.7363e-24
    trafOD                   1.2035      0.092329      13.035    2.6272e-36
    trafOD2               -0.038667     0.0058002     -6.6665    4.0707e-11
    trafOD3              0.00084358    0.00012837      6.5713     7.567e-11
    trafOD4             -5.0231e-06    8.2959e-07     -6.0548    1.9049e-09


1152 observations, 1141 error degrees of freedom
Estimated Dispersion: 19.1
F-statistic vs. constant model: 1.01e+03, p-value = 0
R-squared: 0.8983
```

Table 5-7: Stepwise process for DC2DC service

```
1. Adding trafOD2                       AIC = 5757.7183
2. Adding trafOD4                       AIC = 5609.9179
3. Adding gravity_mult_bus              AIC = 5498.0511
4. Adding dc2                           AIC = 5357.8557
5. Adding trafOD                        AIC = 5320.9379
6. Adding gateway                       AIC = 5241.0584
7. Adding Hour                          AIC = 5206.307
8. Adding Hour5                         AIC = 5081.3997
9. Adding trafOD3                       AIC = 5066.8839
10. Adding cdn2                         AIC = 5048.6734
```

The AIC of the result model, with cross-product terms, began at 5648 and it decreases to 4204. It selected 26 variables and R-squared worth 0.9447.

The Jarque-Beras test reject the normality of the residual distribution. Figure 5-15a represents the normal plot of the residuals and we observe the same thing as *CDN* case. The Ljung-Box Q-test accepts the null hypothesis and confirms that the residuals of this model are independent. As shown in Figure 5-15b, there is no strong autocorrelation among these residuals.

Figure 5-16 shows an example of *DC2DC* traffic flow prediction, where the model made some bad predictions due to the extra flow generated by migration: it underestimated the traffic flow when a migration occurred and overestimated the flow when there was not migration. Nevertheless, it fits well in general since it clearly identifies when the migration of a large volume of data occurs.

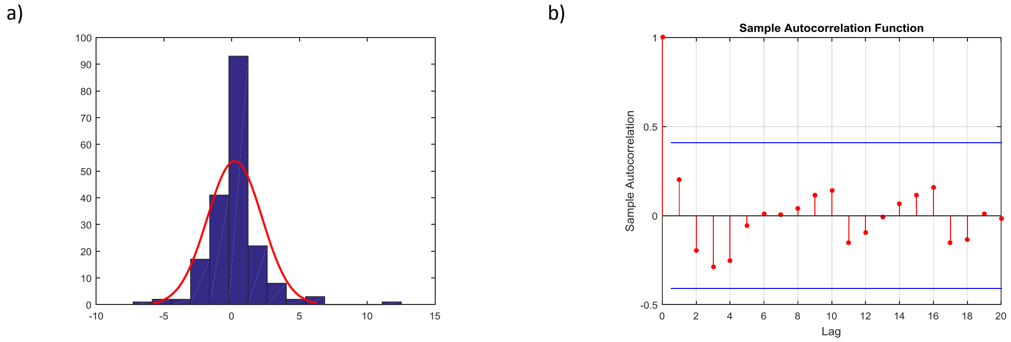


Figure 5-15: Normal plot and autocorrelation for residuals of *DC2DC* model.

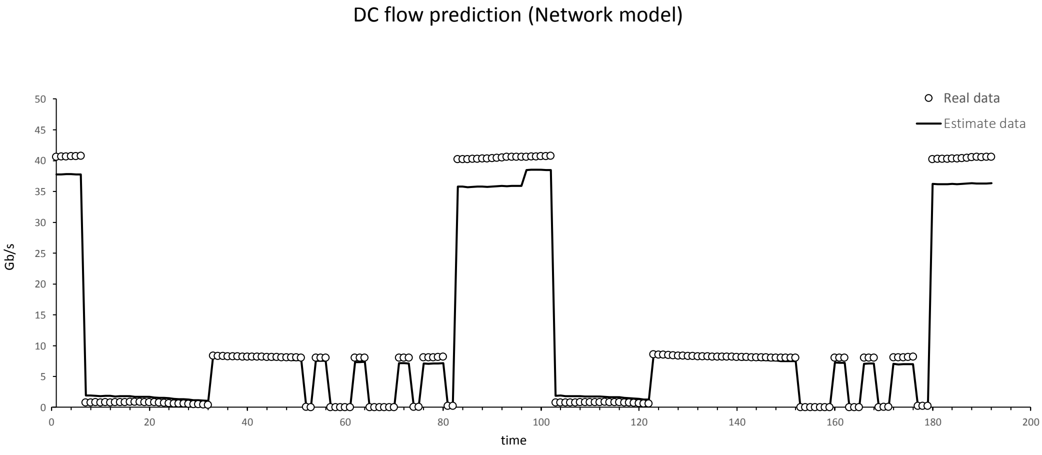


Figure 5-16: *DC2DC* flow prediction (from node 2 to node 4)

As final remark of this section, we conclude that proposed stepwise procedure selects relevant explanatory and provides accurate models. Although the normality tests fails in some cases, we observed few relative errors when comparing predicted and observed values. Therefore, we consider that the methodology used is valid for the proposed case study.

5.5 Modelling approaches evaluation

In this section, we compare the different modelling approaches from three distinct points of view: the overall goodness-of-fit for distinct services, the amount of service traffic monitoring data required to obtain good enough models, and the sensitivity of aggregated traffic flow as explanatory variable,

5.5.1 Goodness-of-fit evaluation

We process the evaluation of the prediction error for the model of different approaches. Figure 5-17 and Figure 5-18 illustrate the *Business* and *DC2DC* prediction errors for different approaches as a function of the time of the monitoring data generation process. Therefore, the higher is the amount of days, the higher is the amount of data used to fit models. Additionally, we assume different percentages of service flow monitoring data availability, a parameter that is comparable to the amount of DPI resources deployed to monitor service traffic flows.

In view of the figures, one can observe that *OD* models take more time to converge than the other two approaches. This is due to the fact that the *OD* models have only data of an OD pair to fit a model, its error will not be established until getting a meaningful amount of monitoring data.

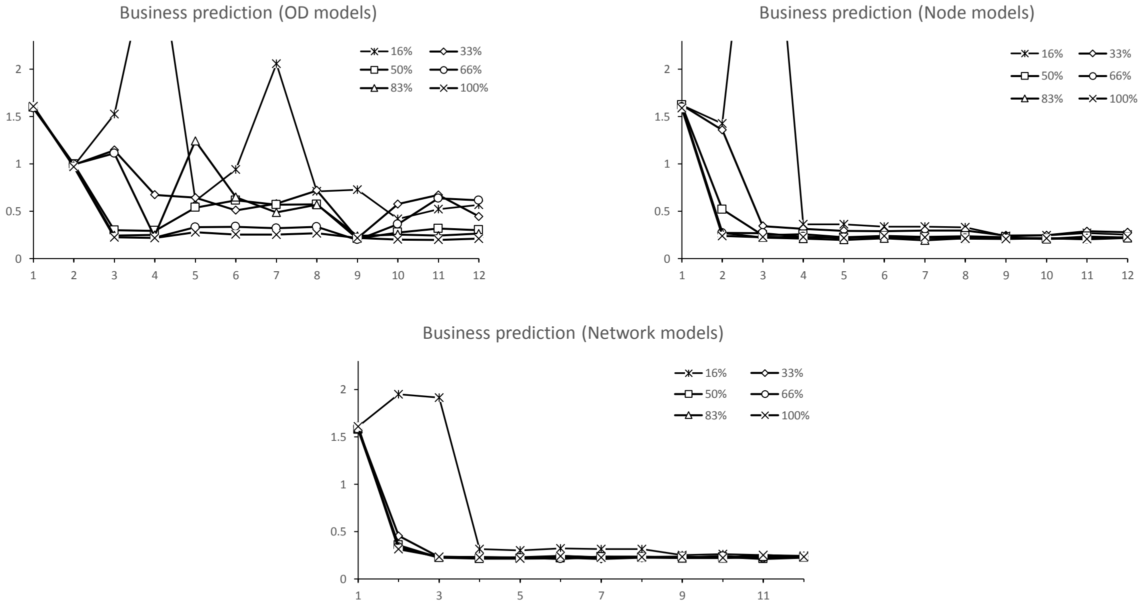


Figure 5-17: Prediction errors of *Business*.

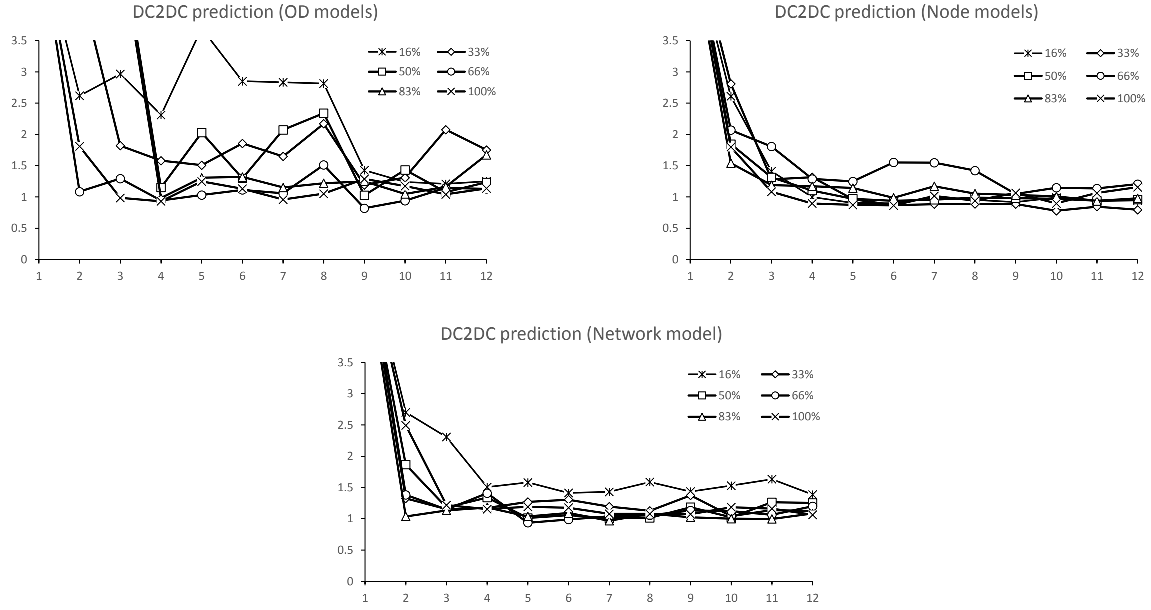


Figure 5-18: Prediction errors of *DC2DC*.

Minimum observed errors for each service are illustrated in Figure 5-19. Both *CDN* and *DC2DC* services have much larger errors than *Business* and *Residential*. The reason behind that fact is that the first two services generate higher traffic than the last two as explained in 5.2. The Figure 5-20 illustrates the relative error of the prediction, it shows that *CDN* has very small relative error, so its big absolute error is due to larger traffic generation. In the same figure, one can observe that *DC2DC* has also large relative error; this fact is because that the largest volume part of *DC2DC* traffic flows is related to data migration of large volumes. In contrast, regular traffic of this service is very small compared with the migration traffic as shown in Figure 5-16. Then, the relative error is very large when no data migration occurs and the model overestimate the traffic flow.

From the previous error figures, main conclusions about distinct approaches are twofold. On the one hand, *Network* and *Node* models converge faster that the OD one, which clearly becomes unappropriated when few monitoring data is available. On the other hand, errors are smaller with the *Node* approach compared with other approaches, except in the case of *Business*, where the error in Node models is 0.22 (being 0.2 in OD models). These results open the door to propose the *Node* approach as the best candidate for the proposed case study. However, a deep analysis (coming in next subsection) is required to finally validate this proposal.

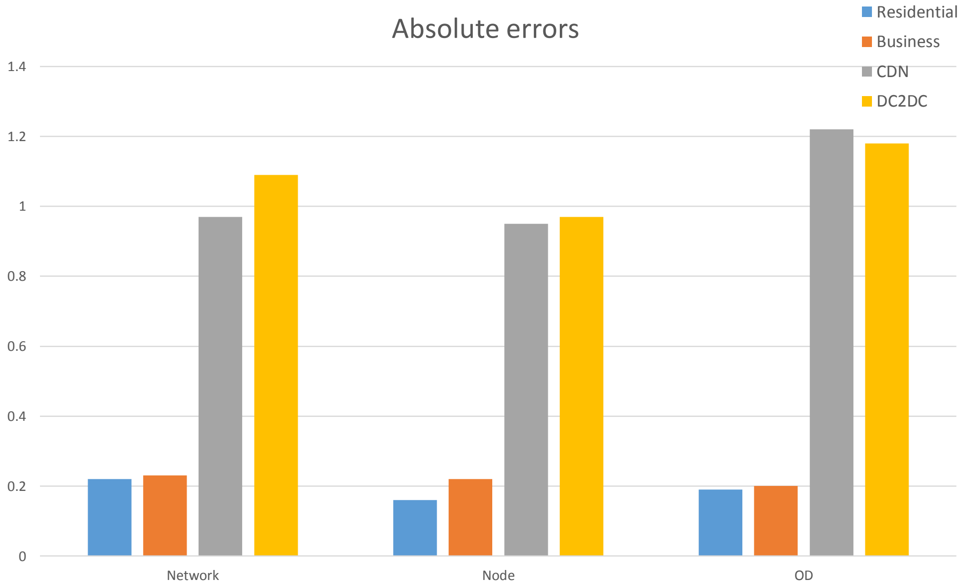


Figure 5-19: Absolute error comparison

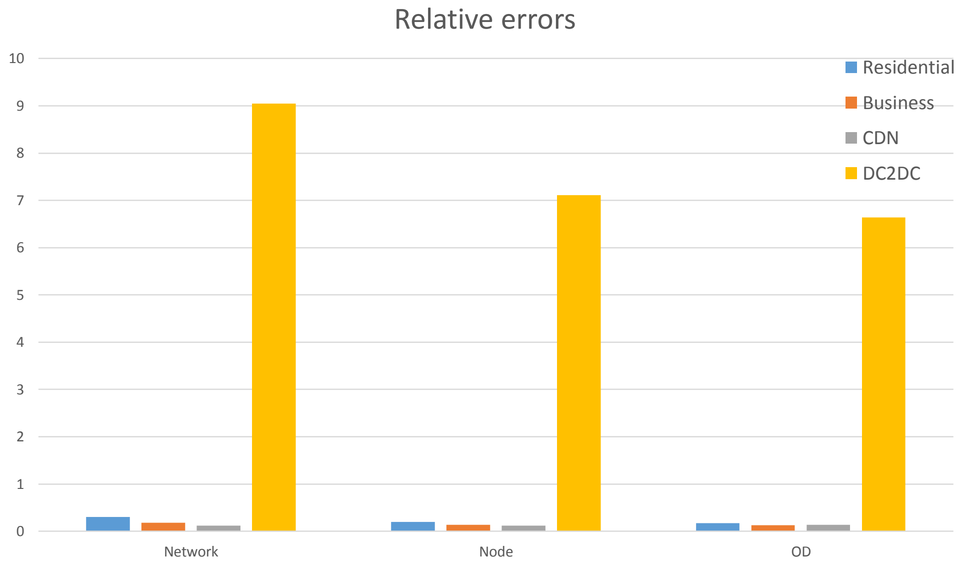


Figure 5-20: Relative error comparison

5.5.2 Analysis of required monitoring data

In the previous section we analyzed the prediction errors for each approaches. As one can observe from Figure 5-17 and Figure 5-18, it is not necessary to deploy resources to monitor 100% of service traffic to reach the minimum error in most of the cases. In this section, we aim at analyzing the necessary volume of data to fit good enough models, i.e. close to a given target error. Recall that this volume is an important factor to analyze since it affects how resources dedicated to service traffic analysis (both computing and storage) need to be configured.

For the sake of a fair comparison, we set a stabilized error as an indicator that the model has a good fitting; this error is obtained doing the average of the errors of the models at the last simulation day (i.e. when all simulated data becomes available for modelling).

After setting the stabilized error as target, the next step is calculate the volume of data that each model requires to reach this error. We define such volume as a dimensionless value obtained by multiplying the percentage of service traffic low monitoring by the required simulation time. Figure 5-21 illustrates the necessary volumes to get the stabilized error with 50% and 100% of service flow monitoring rate. The volumes are obtained by the average of the errors of each services.

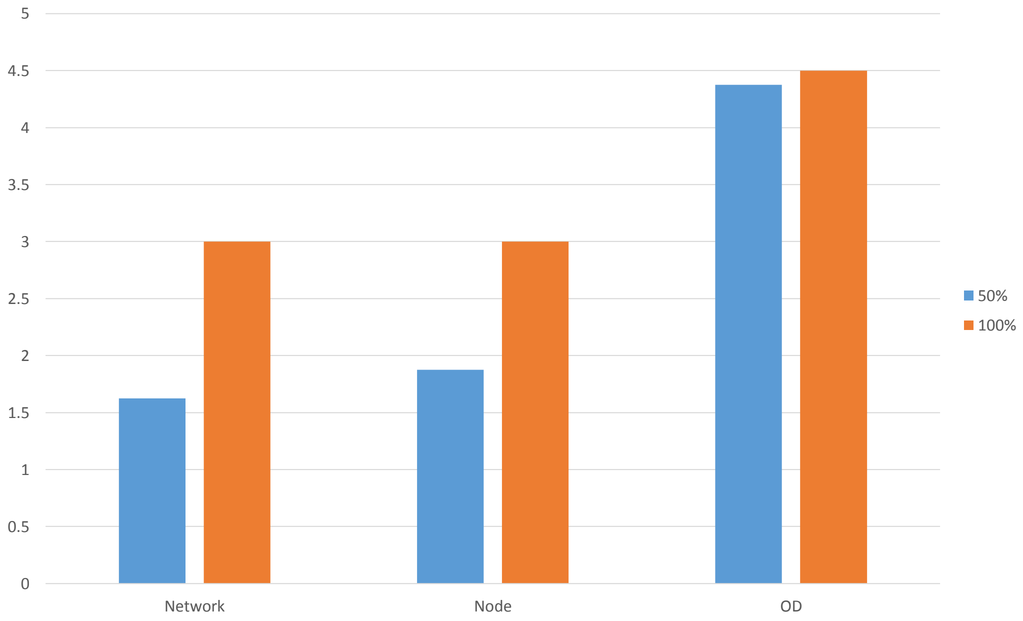


Figure 5-21: Requires data volume for each services

It is worth noting that necessary volumes reduce significantly using 50% of monitoring data when *Network* and *Node* approaches are used. On the contrary, no relevant differences are observed for the OD approach when using more or less resources. However, this approach requires much more data volume than the others to get good fitting.

The reduction of required data by using 50% of DPI resources instead of 100% is shown in Table 5-8. The largest reduction is observed under the *Network approach*, decreasing 46% of the necessary volume; *Node* models reduce 38% and the *OD* models have only 3% of reduction. The average reduction, which is just the average of the three approaches reduction values, is 29%; with this, we can conclude that modelling with 50% of resources reduce the volume of data we need to analyze, what means saving 29% of DPI resources.

Table 5-8: Required data volumes and relative reduction

	50%	**100%**	**reduction**
Network	1.625	3	46%
Node	1.875	3	38%
OD	4.375	4.5	3%
avg			**29%**

Comparing *Network* and *Node* approaches, the former seems to be more efficient since it needs slightly smaller data volumes for 50%. However, its practical implementation in a real network would require centralizing lots of monitoring data, whereas the *Node* approach would allow obtaining models close where data is collected and stores thus, reducing the total amount of network traffic overhead. Therefore, the *Node* approach becomes again the more appropriated solution for the proposed case study.

5.5.3 Sensitivity analysis of *trafOD*

Every of the final models of the proposed approaches includes variable *trafOD* as significant factor. It means that, the estimation of a given service flow for a given OD pair at a given time depends on the aggregated traffic flow of such OD pair. This is useful to estimate the amount of service flows by means of aggregated traffic flow as unique available traffic monitoring data. However, when the objective of model application is to predict future service traffic, it is worth noting *trafOD* variable is unknown for that time, unless some existing model could be used to anticipate such aggregated traffic (e.g. those used in [Mo16]).

To evaluate the impact of such important explanatory variables, we evaluate two alternatives that explicitly exclude such variable: *i*) models without *trafOD*, and *ii)* models considering a variant of *trafOD* called *prevTrafOD* that holds for the amount of aggregated traffic measured in the last period. Note that both of them can be used for predicting future values since all variables are perfectly known in advance. Evaluation is done by comparing the errors of these two new alternatives with respect to the errors of model containing *trafOD*.

Figure 5-22 shows the errors of the modelling with three different set of explanatory variables: with *trafOD*, without *trafOD*, and with *prevTrafOD*. Obviously, errors are smallest when modelling with *trafOD*; it increases when modelling with *prevTrafOD*, which have an approximate information about *trafOD*; On the contrary, modelling without *trafOD* has the biggest error since it has no information about aggregated flows.

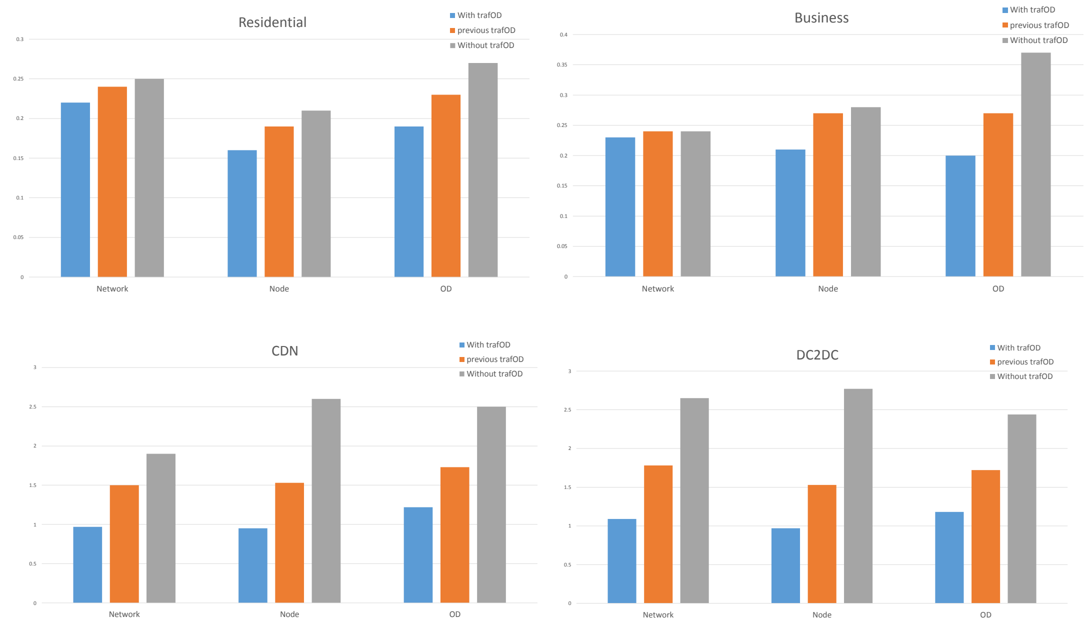


Figure 5-22: Errors of the models for different set explanatory variables.

For *Residential* and *Business* services, errors do not behave significantly different under the *Network* approach. In contrast, the errors have grown considerably in the other approaches, especially for the case of *OD* when modelling without *trafOD*. This behavior is due to the fact that *OD* models only have *hour* and *trafOD* while the other two approaches use also other information, such as distance and the number of users of each service, as explanatory variables. Therefore, the information of *trafOD* is more relevant for *OD* models than for the rest of approaches.

For *CDN* and *DC2DC* services, errors are much larger when the modelling is without *trafOD*. This is because these two services has a strong linear relation with *trafOD*, so predicted values are strongly dependent on this variable. Therefore, the errors with the modelling without *trafOD* largely increase compared to other sets of explanatory variables.

Finally we concluded that the modelling without *trafOD* at the same time step makes bigger errors (larger in the case of modelling without *trafOD*). This increment of errors depends on the different services and different approaches: in some cases the error increase is small like the cases of *Residential* and *Business* models under *Network* approach, whereas it considerably grows for the models of *CDN* and *DC2DC*.

5.6 Summary

In this chapter we evaluated the utility of the developed simulation platform (combining realistic traffic generation and service flows modelling) by means of a case study, where distinct modelling approaches namely, *Network*, *Node*, and *OD* have been studied. Ranging from centralized to distributed data analytics architectures, approaches have been designed to illustrate the applicability of our developed tool to fit a wide range of real deployments.

From the numerical results, we concluded that the *Node* approach becomes the most appropriate one for obtaining service flow models meeting desirable low target errors. Models include a variate set of explanatory variables including characteristics of source and destination locations as well as current aggregated traffic. Additionally, *Node* models experience fast converge to reach that target error as soon as the amount of available service traffic monitoring data increases. Moreover, that data can be obtained allowing a great reduction of the amount of monitoring resources without compromising the quality of models.

Finally, an analysis of getting rid of the current aggregated traffic flow as explanatory variable has been carried out to demonstrate the significance of such variable and to illustrate the poorer goodness-of-fit of such reduced models. Nevertheless, using aggregated traffic monitored at the previous time step clearly improves traffic prediction accuracy thus, mitigating the impact of unknown current aggregated traffic, which occurs when models are used for service traffic forecasting.

Chapter 6.

Concluding Remarks

6.1 Contributions and conclusions

This project focused on the topic of service traffic flow modelling in the telecom cloud. Starting from previous works on modelling aggregated OD traffic flows, the contributions of this project were: *i*) the development of a statistical modelling procedure to obtain service traffic flow models from a wide range of heterogeneous explanatory variables; and *ii*) the design and implementation of a simulation platform able to generate realistic service traffic flows, which also integrates the traffic flow modelling procedures to evaluate distinct approaches for traffic monitoring, data collection, and traffic modelling.

It is important to recall that the current implemented simulator separates the traffic generation from service flow modelling. Therefore, it allows many possibilities to configure a simulation, where one could combine the modelling with different data analytics approaches by selecting respective information from data monitoring for each approach and choosing the modelling methods in the modelling block. Thus, the resultant simulator becomes a flexible tool that can easily adapt to different technologies and scenarios.

The utility of the tool has been demonstrated by means of a case study that was designed to compare distinct data analytics approaches ranging from centralized to distributed approaches. From the numerical results obtained, analysis on several key performance metrics such as models goodness-of-fit, amount of data to be monitored/collected, and impact of key explanatory variables was carried out. Main conclusions are aligned with the idea that combining the benefits of centralized and distributed data analytics architectures become the most appropriate solution for accurate traffic prediction.

6.2 Personal Evaluation

For the achievement of this project, I have been able to apply many of knowledge that I had studied during the degree and the Master, such as statistical modelling, time series, etc. During my personal training, I had great interest in coding and algorithmic methods, especially in mathematical modelling with Matlab that I learnt from *Numerical methods in ODEs*. After the degree, I chose the two Numerical methods course in the Master: *Numerical methods with PDEs* and *Numerical methods for Dynamic Systems*, to improve the knowledge about numerical methods, where I acquired a strengthening in Matlab and what is really helpful for the construction of the simulator.

In this project, I learnt about telecom cloud infrastructures, network traffic analytics and data analysis by means of reading specific papers and the valuable help of my advisors. This project also gave me an opportunity to see how a university researchers group works: discuss of the problems they have confront, the project organization, the elaboration of a paper for some conference, etc. which have provided an enriched experience for me in the area of research.

I want to thank my advisors, Luis Velasco and Marc Ruiz, who acted as mentors and friends to me during this project, they have not been only help me in the project issues but also give me advices for my professional career. It has been a delight to work alongside them in this project.

6.3 Future Work

In this project, linear regression has been proved as valuable technique for modelling service traffic flows. However, there are other techniques that could be evaluated (e.g. neural networks) in order to improve the prediction of some services.

One important future way to explore is to use a real data set to substitute traffic generation and evaluate distinct approaches in the context of a real operator. To this aim, we propose to contact with telecom operators such as Telefonica or British Telecom (which are partners in projects where the research group is enrolled) to ask for such kind of monitoring data. Since a large and complete data set is really difficult to be obtained, another approach to follow is to adjust and improve our traffic generation models from real sparse monitoring measures. In this way, artificial but close-to-real traffic can be generated in a continuous manner.

Apendix A. Linear Models

Residential

This is the Stepwise process (with interactions as the largest set of terms in the fit) and the network model for *Residential* service traffic:

1. Adding dist	**AIC = 7679.6149**
2. Adding cdn1	AIC = 6830.1726
3. Adding Hour2	AIC = 6331.7569
4. Adding Hour	AIC = 6101.079
5. Adding Hour:Hour2	AIC = 4910.8546
6. Adding res2	AIC = 4560.5894
7. Adding emp2	AIC = 4177.8998
8. Adding res1	AIC = 3891.1787
9. Adding Hour8	AIC = 3637.9929
10. Adding Hour4	AIC = 2903.4197
11. Adding Hour7	AIC = 1629.2175
12. Adding cache	AIC = 1389.0891
13. Adding dist:res2	AIC = 1205.8666
14. Adding Hour2:dist	AIC = 1108.7606
15. Adding dist:cache	AIC = 1018.4288
16. Adding Hour:dist	AIC = 977.7195
17. Adding Hour4:dist	AIC = 801.4322
18. Adding gravity_mult_res	AIC = 758.6633
19. Adding res2:emp2	AIC = 672.6621
20. Adding Hour2:cdn1	AIC = 645.2577
21. Adding Hour:Hour4	AIC = 621.4079
22. Adding res1:emp2	AIC = 607.4067
23. Adding gravity_mult_bus	AIC = 588.4361
24. Adding gravity_mult_bus:emp2	AIC = 564.1461
25. Adding dc1	AIC = 551.2926
26. Adding dist:dc1	AIC = 542.6115
27. Adding Hour:cdn1	AIC = 534.7524
28. Adding Hour4:cdn1	AIC = 492.5239
29. Adding trafOD	AIC = 490.1751
30. Adding dist:gravity_mult_bus	AIC = 488.1088
31. Adding gateway	AIC = 485.3541
32. Adding dist:res1	AIC = 481.1013
33. Adding Hour2:res2	AIC = 479.1049
34. Adding Hour9	AIC = 412.0921
35. Adding dist:emp2	AIC = -120.8881
36. Adding dist:cdn1	AIC = -558.1568
37. Adding Hour7:dist	AIC = -676.9325
38. Adding dc1:emp2	AIC = -747.5886

39. Adding gravity_mult_res:emp2	AIC = -790.6593
40. Adding gateway:res2	AIC = -968.9476
41. Adding cdn2	AIC = -1007.6695
42. Adding gravity_mult_res:gateway	AIC = -1074.2102
43. Adding emp1	AIC = -1122.5286
44. Adding Hour2:gateway	AIC = -1132.4877
45. Adding gravity_mult_bus:cache	AIC = -1134.5813
46. Adding emp1:res2	AIC = -1141.1393
47. Adding gravity_mult_res:trafOD	AIC = -1144.305
48. Adding trafOD6	AIC = -1146.2113
49. Adding emp2:trafOD	AIC = -1150.1205
50. Adding gateway:trafOD	AIC = -1153.568
51. Adding cdn2:trafOD	AIC = -1156.2491
52. Adding Hour4:trafOD	AIC = -1158.0296
53. Adding Hour:dc1	AIC = -1158.9814
54. Adding Hour:res2	AIC = -1159.8863
55. Adding Hour4:res2	AIC = -1167.4825
56. Adding Hour:gravity_mult_res	AIC = -1169.2334
57. Adding Hour2:gravity_mult_res	AIC = -1171.9401
58. Adding Hour:res1	AIC = -1174.1146
59. Adding Hour4:gravity_mult_res	AIC = -1175.7706
60. Adding dc1:trafOD	AIC = -1177.5009
61. Adding Hour:cache	AIC = -1177.5288
62. Adding Hour2:cache	AIC = -1180.8955
63. Adding Hour4:cache	AIC = -1190.6818
64. Adding Hour9:cache	AIC = -1200.4428
65. Adding dc1:res2	AIC = -1200.4823
66. Adding Hour2:gravity_mult_bus	AIC = -1200.6226
67. Removing gravity_mult_res:trafOD	AIC = -1202.5
68. Removing gateway:trafOD	AIC = -1204
69. Removing Hour2:cdn1	AIC = -1205.6
70. Removing Hour:cdn1	AIC = -1207
71. Removing res1:emp2	AIC = -1208.2
72. Removing Hour4:trafOD	AIC = -1208.9
73. Removing dc1:emp2	AIC = -1209.5
74. Removing dc1:res2	AIC = -1211.2
75. Removing dist:gravity_mult_bus	AIC = -1211.7
76. Adding cdn1:res2	AIC = -1212.4139
77. Removing Hour2:gateway	AIC = -1212.5
78. Adding Hour9:gateway	AIC = -1213.1864

```
Generalized Linear regression model:
    Residencia ~ [Linear formula with 59 terms in 20 predictors]
    Distribution = Normal

Estimated Coefficients:
                                 Estimate          SE         tStat        pValue
               __________    _________    _______    _________

    (Intercept)                     0            0            0             0
    Hour                        0.728     0.512334       13.173             0
    Hour2                       0.351     0.010392       33.776   2.5617e-210
    Hour4                    0.032981   0.00049702       66.357             0
    Hour7                  5.6751e-06   8.3856e-08       67.676             0
    Hour8                 -2.4108e-07   3.7082e-09      -65.014             0
    Hour9                  3.1949e-09   5.1414e-11        62.14             0
    dist                   -0.0027223   0.00035961        -7.57    5.0237e-14
    gravity_mult_res      -0.00011596   2.1905e-05      -5.2938    1.2898e-07
    gravity_mult_bus       3.0371e-05   1.3439e-06         22.6   4.4343e-104
    gateway                    2.0012   0.00049702       5.5753    5.2621e-07
    cache                     -1.8348   0.03029502       5.5753    1.1729e-06
    res1                   0.00064325   0.00011537       5.5753    2.7054e-08
    emp1                         -0.00097939    8.0065e-05     -12.233     1.4538e-33
    cdn1                           0.0012445    0.00025086      4.9611     7.4244e-07
    dc1                          -8.1596e-05    1.5865e-05     -5.1431     2.8869e-07
    res2                          0.0010654    0.00023342     -4.5645     5.2219e-06
    emp2                          -0.0019781    5.2668e-05     -37.558    1.0522e-250
    cdn2                          0.00056095    4.6036e-05      12.185     2.5336e-33
    trafOD                         0.0025008     0.0011487      2.1771       0.029556
    trafOD6                       9.0611e-13    1.8597e-13      4.8725     1.1629e-06
    Hour:Hour2                      -0.19632     0.0035278     -55.648              0
    Hour:Hour4                    -0.0020544    2.9415e-05     -69.843              0
    Hour:dist                    -0.00085039    3.1478e-05     -27.016    3.7272e-143
    Hour:gravity_mult_res          8.079e-07    1.4641e-07      5.5179     3.7417e-08
    Hour:cache                   -2.8348e-09    1.4103e-06           0              0
    Hour:res1                    -3.0849e-06    1.3565e-06     -2.2742       0.023029
    Hour:dc1                      3.0604e-07    1.3531e-07      2.2618       0.023786
    Hour:res2                    -2.4944e-05    2.7011e-06     -9.2347      4.945e-20
    Hour2:dist                    7.5479e-05    2.9686e-06      25.426    1.4049e-128
    Hour2:gravity_mult_res       -5.0992e-08    1.0552e-08     -4.8326     1.4194e-06
    Hour2:gravity_mult_bus        1.3425e-09    5.6167e-10      2.3903       0.016901
    Hour2:cache                   0.00023854    0.00034901     0.68348        0.49436
    Hour2:res2                    1.6606e-06    1.8655e-07      8.9015     9.6487e-19
    Hour4:dist                   -1.1834e-07    6.2024e-09      -19.08     1.9782e-76
    Hour4:gravity_mult_res        4.4502e-11    1.1301e-11      3.9377     8.4263e-05
    Hour4:cache                    -5.67e-07    1.1697e-06    -0.48474         0.6279
    Hour4:cdn1                   -3.7654e-10    1.5912e-10     -2.3665       0.018027
    Hour4:res2                   -1.7358e-09    2.3491e-10     -7.3893     1.9347e-13
    Hour7:dist                    3.3293e-12    2.5117e-13      13.255     6.0191e-39
    Hour9:gateway                 1.0108e-13    5.3612e-14      1.8854       0.059475
    Hour9:cache                   7.5568e-15    1.0604e-13    0.071267        0.94319
    dist:cache                     0.0019421    0.00026535      7.3189     3.2449e-13
    dist:res1                    -4.9166e-06    4.4041e-07     -11.164     2.3889e-28
    dist:cdn1                     5.6829e-06    3.6585e-07      15.533     2.8352e-52
    dist:dc1                     -2.3905e-07    1.6727e-08     -14.291     8.7828e-45
    dist:res2                    -1.3236e-06    3.3915e-07     -3.9026     9.7396e-05
    dist:emp2                     1.2518e-06    1.1877e-07       10.54      1.679e-25
    gravity_mult_res:gateway      0.00028036    2.0347e-05      13.779     7.4725e-42
    gravity_mult_res:emp2        -2.4268e-08    1.2926e-09     -18.775     3.3707e-74
    gravity_mult_bus:cache        4.0073e-06    9.1836e-07      4.3636     1.3254e-05
    gravity_mult_bus:emp2        -2.8348e-09    2.6319e-10     -10.771     1.5443e-26
    gateway:res2                  -0.0081738    0.00052643     -15.527     3.1106e-52
    emp1:res2                    -7.1688e-08    1.3437e-08     -5.3353     1.0293e-07
    cdn1:res2                     6.3127e-08    3.5549e-08      1.7758       0.075879
    dc1:trafOD                   -6.5895e-07    1.2718e-07     -5.1812     2.3595e-07
    res2:emp2                     1.4939e-06    9.4125e-08      15.871     2.1141e-54
    emp2:trafOD                   3.0415e-06    5.7937e-07      5.2496     1.6373e-07
    cdn2:trafOD                  -3.6219e-06    8.5479e-07     -4.2372     2.3355e-05


2880 observations, 2826 error degrees of freedom
Estimated Dispersion: 0.0326
F-statistic vs. constant model: 8.8e+03, p-value = 0
R-squared: 0.9940
```

Business

This is the Stepwise process (with interactions as the largest set of terms in the fit) and the network model for *Business* service traffic:

```
1. Adding emp1                               AIC = 6982.3719
2. Adding emp2                               AIC = 6356.3868
3. Adding Hour                               AIC = 6045.0455
4. Adding Hour4                              AIC = 4512.3262
5. Adding dc2                                AIC = 3958.2885
6. Adding dc1                                AIC = 3338.4896
7. Adding dist                               AIC = 3104.7483
8. Adding gateway                            AIC = 2793.4826
9. Adding emp1:emp2                          AIC = 2538.015
10. Adding Hour2                             AIC = 2392.249
11. Adding Hour3                             AIC = 737.5422
12. Adding Hour:emp1                         AIC = 625.6696
13. Adding Hour4:emp1                        AIC = 199.8091
14. Adding gravity_mult_bus                  AIC = 77.8897
15. Adding cache                             AIC = 9.6351
16. Adding dist:cache                        AIC = -35.3595
17. Adding dist:dc1                          AIC = -109.1556
18. Adding dc1:emp2                          AIC = -151.0136
19. Adding Hour:emp2                         AIC = -188.416
20. Adding Hour4:emp2                        AIC = -331.142
21. Adding gravity_mult_bus:dc1              AIC = -354.5485
22. Adding cdn1                              AIC = -401.579
23. Adding dist:cdn1                         AIC = -517.6986
24. Adding Hour:Hour4                        AIC = -539.5044
25. Adding Hour6                             AIC = -828.2307
26. Adding Hour2:Hour6                       AIC = -1588.2748
27. Adding Hour2:emp1                        AIC = -1620.1102
28. Adding Hour3:emp1                        AIC = -1831.0254
29. Adding cache:dc2                         AIC = -1865.3417
30. Adding cdn1:dc2                          AIC = -1908.7964
31. Adding res2                              AIC = -1941.7344
32. Adding Hour:dc2                          AIC = -1958.9471
33. Adding Hour3:dc2                         AIC = -2027.6304
34. Adding Hour:dc1                          AIC = -2046.9708
35. Adding Hour4:dc1                         AIC = -2105.6345
36. Adding gravity_mult_bus:emp2             AIC = -2119.7752
37. Adding dc1:dc2                           AIC = -2129.9363
38. Adding emp2:dc2                          AIC = -2148.0017
39. Adding dist:emp1                         AIC = -2187.3304
40. Adding cdn2                              AIC = -2213.4554
41. Adding gravity_mult_bus:emp1             AIC = -2222.8921
42. Adding gateway:cache                     AIC = -2230.6518
43. Adding Hour2:emp2                        AIC = -2238.1237
44. Adding Hour3:emp2                        AIC = -2288.211
45. Adding Hour2:dc2                         AIC = -2296.2314
46. Adding Hour4:dc2                         AIC = -2305.6038
47. Adding Hour:dist                         AIC = -2309.9415
48. Adding Hour4:dist                        AIC = -2334.5298
49. Adding Hour:gravity_mult_bus             AIC = -2336.8258
50. Adding Hour4:gravity_mult_bus            AIC = -2345.7651
51. Removing gravity_mult_bus:emp2           AIC = -2345.8
52. Adding gravity_mult_bus:dc2              AIC = -2346.0756
53. Adding Hour2:gateway                     AIC = -2346.2849
54. Adding Hour6:gateway                     AIC = -2369.4733
55. Adding Hour2:dist                        AIC = -2375.9499
56. Adding Hour3:dc1                         AIC = -2377.1134
57. Adding Hour2:dc1                         AIC = -2398.7069
58. Removing Hour:dist                       AIC = -2400.7
```

```
Generalized Linear regression model:
    Business ~ [Linear formula with 55 terms in 16 predictors]
    Distribution = Normal

Estimated Coefficients:
                              Estimate         SE         tStat       pValue
                            ___________   __________   _________   ___________

    (Intercept)                  -3.7498     0.050062     -74.903             0
    Hour                         0.30671     0.024549      12.494    6.6351e-35
    Hour2                       -0.41364     0.011888     -34.794   4.5332e-221
    Hour3                        0.12073    0.0026321      45.868             0
    Hour4                      -0.013826   0.00028877     -47.879             0
    Hour6                    -1.6574e-05   3.6988e-07      -44.81             0
    dist                      -0.0041696   9.5668e-05     -43.583    8.081e-318
    gravity_mult_bus         -3.0872e-05   6.2679e-06     -4.9254    8.9032e-07
    gateway                      -1.2894   1.5855e-05           0             0
    cache                        0.65678     0.096953      6.7742    1.5163e-11
    emp1                      0.00049783   3.4815e-05      14.299    7.8363e-45
    cdn1                      0.00017747   5.8951e-05      3.0105     0.0026311
    dc1                      -7.1287e-05   5.2939e-06     -13.466    4.1836e-40
    res2                      0.00026441   3.8125e-05      6.9354    5.0021e-12
    emp2                       0.0010233   1.7689e-05      57.848             0
    cdn2                     -0.00031162   4.3372e-05     -7.1848    8.5704e-13
    dc2                      -2.1316e-05   5.5636e-06     -3.8313    0.00013025
    Hour:Hour4                0.00074047   1.5855e-05      46.703             0
    Hour:gravity_mult_bus     1.3277e-07   2.5415e-08      5.2242    1.8762e-07
    Hour:emp1                  4.678e-05    3.442e-06      13.591    8.4641e-41
    Hour:dc1                 -6.0954e-06   1.6083e-06       -3.79    0.00015379
    Hour:emp2                 2.2055e-05    3.442e-06      6.4075    1.7274e-10
    Hour:dc2                 -7.0283e-06   1.6083e-06     -4.3701    1.2869e-05
    Hour2:Hour6               3.8279e-09    9.251e-11      41.379   5.4929e-293
    Hour2:dist                5.1403e-06   5.6653e-07      9.0733    2.1092e-19
    Hour2:gateway              0.0023767   0.00046319       5.131    3.0765e-07
    Hour2:emp1               -1.2196e-05   6.2301e-07     -19.576    4.0307e-80
    Hour2:dc1                 1.5531e-06   2.9359e-07      5.2901    1.3163e-07
    Hour2:emp2               -5.9992e-06   6.2301e-07     -9.6294    1.2859e-21
    Hour2:dc2                 1.7983e-06   2.9359e-07      6.1254    1.0304e-09
    Hour3:emp1                7.2177e-07   4.2997e-08      16.787    2.3384e-60
    Hour3:dc1                 -1.063e-07    1.948e-08     -5.4567      5.27e-08
    Hour3:emp2                3.2863e-07   4.2997e-08      7.6431    2.8847e-14
    Hour3:dc2                -1.2115e-07    1.948e-08     -6.2191    5.7401e-10
    Hour4:dist               -1.0578e-08   1.1585e-09     -9.1306     1.263e-19
    Hour4:gravity_mult_bus   -9.8641e-12   2.1832e-12     -4.5182     6.492e-06
    Hour4:emp1               -1.1996e-08     1.01e-09     -11.877    8.7327e-32
    Hour4:dc1                 2.2118e-09   4.2461e-10      5.2091    2.0343e-07
    Hour4:emp2               -4.5941e-09     1.01e-09     -4.5485    5.6296e-06
    Hour4:dc2                 2.4636e-09   4.2461e-10       5.802     7.279e-09
    Hour6:gateway            -1.1246e-08   1.9346e-09     -5.8132    6.8157e-09
    dist:cache                -0.0028659   0.00029389     -9.7518    4.0288e-22
    dist:emp1                 1.1356e-06   5.5589e-08      20.429    1.2567e-86
    dist:cdn1                -1.0115e-06   1.5271e-07     -6.6234    4.1865e-11
    dist:dc1                 -2.2791e-08   1.3742e-08     -1.6585      0.097332
    gravity_mult_bus:emp1     8.1997e-09   1.7008e-09      4.8211    1.5035e-06
    gravity_mult_bus:dc1      -3.442e-09    7.492e-10     -4.5942    4.5338e-06
    gravity_mult_bus:dc2      1.1451e-10   2.4352e-10     0.47021       0.63824
    gateway:cache               -0.54906      0.11706     -4.6904    2.8559e-06
    cache:dc2                 4.1486e-05   5.6041e-06      7.4027    1.7509e-13
    emp1:emp2                -1.2894e-09   1.6571e-08   -0.077809       0.93799
    cdn1:dc2                  1.5697e-08   2.3092e-09      6.7976    1.2929e-11
    dc1:emp2                  2.1912e-08   1.7335e-09      12.641    1.1421e-35
    dc1:dc2                   -4.901e-09   8.6111e-10     -5.6915    1.3884e-08
    emp2:dc2                 -4.5667e-08   4.1831e-09     -10.917    3.3212e-27


2880 observations, 2826 error degrees of freedom
```

```
Estimated Dispersion: 0.0194
F-statistic vs. constant model: 9.05e+03, p-value = 0
R-squared: 0.9941
```

CDN

This is the Stepwise process (with interactions as the largest set of terms in the fit) and the network model for *CDN* service traffic:

```
1. Adding trafOD                              AIC = 1484.1445
2. Adding Hour7                               AIC = 1415.3004
3. Adding Hour7:trafOD                        AIC = 1267.2889
4. Adding gravity_mult_bus                    AIC = 1228.2792
5. Adding trafOD7                             AIC = 1211.3947
6. Adding Hour                                AIC = 1193.1486
7. Adding Hour9                               AIC = 1164.2719
8. Adding Hour9:trafOD                        AIC = 1150.8435
9. Adding gravity_mult_res                    AIC = 1149.7467
```

```
Generalized Linear regression model:
    CDN ~ 1 + Hour + gravity_mult_res + gravity_mult_bus + trafOD7 + Hour7*trafOD +
Hour9*trafOD
    Distribution = Normal

Estimated Coefficients:
                         Estimate         SE         tStat       pValue
                        ___________    __________    _______    ___________

    (Intercept)                   0             0          0              0
    Hour                   -0.12032      0.016604     -7.246     1.7724e-12
    Hour7                3.2247e-09    1.2865e-09     2.5065       0.012529
    Hour9               -5.0546e-12    2.4521e-12    -2.0613       0.039822
    gravity_mult_res     5.1043e-05    2.9111e-05     1.7534       0.080188
    gravity_mult_bus    -3.3928e-05    1.3076e-05    -2.5947      0.0097639
    trafOD                  0.86636      0.016548     52.354    3.0389e-198
    trafOD7               -1.59e-12    2.7232e-13    -5.8386        9.83e-09
    Hour7:trafOD         3.4956e-10    7.6149e-11     4.5905     5.6821e-06
    Hour9:trafOD        -4.9208e-13    1.5532e-13    -3.1682      0.0016335


480 observations, 471 error degrees of freedom
Estimated Dispersion: 1.68
F-statistic vs. constant model: 5.23e+03, p-value = 0
R-squared: 0.9889
```

DC2DC

This is the Stepwise process (with interactions as the largest set of terms in the fit) and the network model for *DC* service traffic:

```
1. Adding trafOD                                            AIC = 5648.0884
2. Adding cdn1                                              AIC = 5235.6074
3. Adding gravity_mult_bus                                  AIC = 5146.9429
4. Adding gravity_mult_bus:trafOD                           AIC = 4810.0948
5. Adding trafOD2                                           AIC = 4747.6629
6. Adding cdn1:trafOD                                       AIC = 4683.0647
7. Adding Hour                                              AIC = 4646.8256
8. Adding Hour:cdn1                                         AIC = 4610.2072
9. Adding Hour9                                             AIC = 4579.4277
10. Adding dc2                                              AIC = 4562.9524
11. Adding dc2:trafOD2                                      AIC = 4488.016
12. Adding Hour:dc2                                         AIC = 4468.2796
13. Adding gravity_mult_res                                 AIC = 4451.8439
14. Adding Hour:gravity_mult_bus                            AIC = 4443.471
15. Adding Hour9:trafOD                                     AIC = 4418.7739
16. Adding Hour:trafOD                                      AIC = 4324.2232
17. Adding Hour10                                           AIC = 4307.0234
18. Adding cdn1:dc2                                         AIC = 4293.4995
19. Adding Hour8                                            AIC = 4286.9962
20. Adding Hour:gravity_mult_res                            AIC = 4286.1833
21. Adding Hour8:trafOD                                     AIC = 4286.0331
22. Adding gravity_mult_bus:cdn1                            AIC = 4285.9956
23. Removing Hour9:trafOD                                   AIC = 4280.4
24. Adding trafOD7                                          AIC = 4228.6897
25. Adding gravity_mult_res:trafOD                          AIC = 4223.1189
26. Adding res2                                             AIC = 4215.9606
27. Adding gravity_mult_res:gravity_mult_bus                AIC = 4212.0124
28. Adding emp1                                             AIC = 4211.874
29. Removing cdn1:dc2                                       AIC = 4210.1
30. Adding Hour:emp1                                        AIC = 4210.06
31. Removing gravity_mult_bus:cdn1                          AIC = 4208.1
32. Removing gravity_mult_bus:trafOD                        AIC = 4206.3
33. Adding gravity_mult_bus:trafOD2                         AIC = 4204.391
```

```
Generalized Linear regression model:
    DC ~ [Linear formula with 26 terms in 13 predictors]
    Distribution = Normal

Estimated Coefficients:
                                          Estimate         SE          tStat        pValue
                                         __________    __________    _________    __________

    (Intercept)                                   0             0            0             0
    Hour                                 2.7829e-10             0            0             0
    Hour8                               -8.9918e-12    1.1113e-09    -0.008091       0.99355
    Hour9                                2.3291e-11    1.0241e-10      0.22743       0.82013
    Hour10                              -9.7118e-13    2.3677e-12     -0.41017       0.68176
    gravity_mult_res                     0.00010792    2.6269e-05       4.1081    4.2776e-05
    gravity_mult_bus                     7.1316e-05     1.209e-05       5.8985    4.8459e-09
    emp1                                 1.5807e-05    0.00022331     0.070786       0.94358
    cdn1                                 0.00088518    0.00037207       2.3791      0.017522
    res2                                 -0.0012975    0.00017838      -7.2739    6.5269e-13
    dc2                                  0.00072162    5.9182e-05       12.193    3.2794e-32
    trafOD                                -0.091095             0            0             0
    trafOD2                                0.027092      0.002238       12.105     8.496e-32
    trafOD7                             -1.3568e-12    6.4125e-14      -21.159     6.278e-84
    Hour:gravity_mult_res                3.4893e-06    9.9248e-07       3.5158    0.00045593
    Hour:gravity_mult_bus                1.1351e-06    2.3124e-07       4.9088    1.0517e-06
    Hour:emp1                            1.2669e-05    1.6584e-05      0.76392       0.44507
    Hour:cdn1                           -0.00012376    2.9188e-05        -4.24     2.418e-05
    Hour:dc2                            -2.6984e-05    4.4746e-06      -6.0305    2.2139e-09
    Hour:trafOD                           -0.013251     0.0016684      -7.9424    4.7753e-15
    Hour8:trafOD                         6.6212e-12    5.2834e-13       12.532    8.0156e-34
    gravity_mult_res:gravity_mult_bus   -3.1702e-09    4.4453e-10      -7.1316    1.7711e-12
    gravity_mult_res:trafOD              4.8735e-06    9.0492e-07       5.3856    8.7869e-08
    gravity_mult_bus:trafOD2            -6.0371e-08    9.3432e-09      -6.4616    1.5395e-10
    cdn1:trafOD                         -7.1659e-05    1.6314e-05      -4.3924    1.2267e-05
    dc2:trafOD2                         -6.2488e-07    2.0863e-07      -2.9952     0.0028025


1152 observations, 1129 error degrees of freedom
Estimated Dispersion: 10.3
F-statistic vs. constant model: 876, p-value = 0
R-squared: 0.9447
```

Apendix B. References

[Ah14] Ken Aho, D. Derryberry, and T. Peterson., “Model selection for ecologists: the worldviews of AIC and BIC”, Ecology vol 95 pp. 631–636, 2014

[AV16] A. P. Vela, A. Via, M. Ruiz, and L. Velasco, "Bringing Data Analytics to the Network Nodes," accepted in European Conference on Optical Communication (ECOC), 2016.

[CISCO] CISCO blogs: “Visual Networking Index,” https://blogs.cisco.com/

[Co12] L.M. Contreras, V. Lopez, O. González, A. Tovar, F. Muñoz, A. Azanon, J.P. Fernandez-Palacios, J. Folgueira, “Toward cloud-ready transport networks, “ IEEE Commun. Mag., 50, pp. 48–55, 2012.

[Fi14] G. Finnie, “DPI & Traffic Analysis in Networks Based on NFV and SDN,” *White Paper,* Heavy Reading, 2014.

[Gi16] Ll. Gifre, L. M. Contreras, V. Lopez, and L. Velasco, "Big Data Analytics in Support of Virtual Network Topology Adaptability," in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), 2016.

[Gr03] Wayne Grover, “Mesh-based Survivable Transport Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking,” Prentice Hall, 2003.

[Jh15] A. Jha, S. Ray, B. Seaman et al. “Clustering to forecast sparse time-series data”, 31st International Conference on Data Engineering. IEEE pp. 1388-1399, 2015

[Ko14] Eric D. Kolaczyk, and Gábor Csárdi, “Statistical Analysis of Network Data with R”. Springer New York, pp. 162-163, 2014. ISBN: 9781493909834.

[Ma14] M. Malboubi, L. Yang, C. Chuah, P. Sharma, “Intelligent SDN based Traffic (de)Aggregation and Measurement Paradigm (iSTAMP),” IEEE

INFOCOM, 2014.

[Mo16] F. Morales, M. Ruiz, and L. Velasco, "Virtual Network Topology Reconfiguration based on Big Data Analytics for Traffic Prediction," in Proc. IEEE/OSA Optical Fiber Communication Conference (OFC), 2016.

[Ne72] J. A. Nelder.and R. J. Baker. "Generalized linear models", In Encyclopedia of statistical sciences, 1972

[NFV] ETSI GS NFV 001: Network Functions Virtualization (NFV): Use Cases, V1.1.1, ETSI, October 2013.

[ONF16] Open Networking Foundation (ONF). www.opennetworking.org (accessed 02/2016).

[Ra95] C. R Rao and H. Toutenburg. "Linear models". Springer New York. pp. 3-18

[Ro11] K.Roebuck, *Deep Packet Inspection: High-impact Strategies - What You Need to Know,* Emereo Pty Limited, 2011.

[Ru16] M. Ruiz, M. Germán, L. M. Contreras, and L. Velasco, "Big Data-backed Video Distribution in the Telecom Cloud," Elsevier Computer Communications, vol. 84, pp. 1-11, 2016.

[SS] SlideShare: In-home Internet Usage Measurement - Truong Si Anh http://www.slideshare.net/Zing_Ads/inhome-internet-usage-measurement-truong-si-anh

[Ts10] R. S. Tsay. *Analysis of Financial Time Series*. 3rd ed. Hoboken, NJ: John Wiley & Sons, Inc., 2010.

[Ve15] L. Velasco, L.M. Contreras, G. Ferraris, A. Stavdas, F. Cugini, M. Wiegand, J.P. Fernández-Palacios, "A service-oriented hybrid access network and cloud architecture," IEEE Commun. Mag., 53, pp. 159–165, 2015.

[We94] Wei, W. W. S. "Time series analysis". Reading: Addison-Wesley publ, 1994.

[Wi16] Coefficient of determination. (2016, June 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 11:48, June 17, 2016, from https://en.wikipedia.org/w/index.php?title=Coefficient_of_determination&oldid=724216387

[Wi16.2] Deep packet inspection. (2016, May 31). In *Wikipedia, The Free Encyclopedia.* Retrieved 21:40, July 11, 2016, from https://en.wikipedia.org/w/index.php?title=Deep_packet_inspection&ol

did=722999084

[Wi99] J. Wisniak and A. Polishuk, " Analysis of residuals — a useful tool for phase equilibrium data analysis" Fluid Phase Equilibria, 1999, 164(1): 61-82.