UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Magistrale in Ingegneria Informatica

An AI based system for supporting semi automatic insurance risk analysis and management

Relatore:

Prof. Carlo Ferrari

Laureando:

Marco Serpelloni

Matricola N. 1105770

Anno Accademico 2022-2023

Data di laurea: dicembre 2023

Contents

Abstract

Day by day, Artificial Intelligence has gained a wide interest to ground itself into business and commercial use. As the influence of technology is growing and the economy shifting to a more digitalized system, this thesis examines how AI can be use in a business field and how can be integrated on a CRM with an innovate concept.

This thesis activity has been started thanks to an interneship between University of Padua and *Oman Insurance Company* that is one of the biggest insurance company on the United Arab Emirates. The IT departemnt of Oman Insurance Company is always at work to improve IT systems and softwares used by all departments of the company. The context of this thesis activity it is a project of a new CRM web application that should manage insurance / reinsurance commercial business activity as far as aviation and space business is concerned.

This project is focused on the following items:

- review of the business data flow and business procedures;
- design a new database, import past data;
- design an AI system that helps users during the evaluation phase of a Business Case;
- design a system that prevents data entry error through an hybrid approach which uses both an AI model and a probabilistic model working together;
- integrate this systems on a modern CRM realized by microservices architecture;

Abstract in italiano

Giorno dopo giorno, l'intelligenza artificiale ha acquisito un vasto interesse nel suo uso aziendale e commerciale. Questa tesi esamina come l'intelligenza artificiale possa essere utilizzata in un campo aziendale e come può essere integrata in un CRM con un concetto innovativo.

Questa attività di tesi è stata avviata grazie ad uno stage tra l'Università di Padova e *Oman Insurance Company* che è una delle più grandi compagnie assicurative degli Emirati Arabi Uniti. Il dipartimento IT di Oman Insurance Company è sempre al lavoro per migliorare i propri sistemi IT e i software utilizzati da tutti i dipartimenti dell'azienda. Il contesto di questa attività di tesi è un progetto di una nuova applicazione web CRM che dovrebbe gestire l'attività commerciale assicurativa/riassicurativa per quanto riguarda il settore aeronautico e spaziale.

Questo progetto è incentrato sui seguenti elementi:

- revisione del flusso dei dati aziendali e delle procedure aziendali;
- progettazione di un nuovo database e importazione dei dati pregressi;
- progettazione di un sistema di AI che aiuti gli utenti durante la fase di valutazione di un affare assicurativo;
- progettare un sistema che prevenga errori di immissione dei dati attraverso un approccio ibrido che utilizza sia un modello di intelligenza artificiale che un modello probabilistico che lavorino insieme;
- integrare questi sistemi su un moderno CRM realizzato mediante architettura a microservizi;

Preface

This thesis work has been developed during the stage at Oman Insurance Company (recently renamed as Sukoon Insurance Company) following the envisaged procedures by University of Padua.

Prof. Carlo Ferrari, Associate Professor of Computer Science at the Department of Electronics and Informatics (DEI) of the University of Padua, acted as the university supervisor whilist Lorenzo Signor, SVP - Head of Aviation Space at Sukoon Company, acted as the company tutor.

The goal of this internship was focused on redesigning the business process applicable to aviation and space insurance risks with a particolar emphasis on 2 points: a way to help users during a specific business phase called Underwriting and a system to prevent data entry errors.

This thesis has been considered like a specific single project. All different activities have been identified and planned as to work closely with the Aviation and Space commercial business unit.
Most of these activities have delivered a milestone that has been discussed and approved both by the IT department and the commercial business unit.

Introduction

Companies are always looking for the most innovative project once it comes to improve their business. Thanks to AI, the advancement of a self-sufficient program, a system with cognitive capabilities, is on the rise in the developed Countries. The effects of modernization and automation of the economy will change the foundation of how businesses will evolve.
Amazon CEO, Jeff Bezos stated that he believes we have entered the *"golden age"* of AI that allows us to solve the problems that once were the realm of sci-fi (Marr, 2019). Google co-founder Sergey Brin is another representative of the development of Artificial Intelligence by expressing that "AI is the most significant development in computing in my life" and Microsoft CEO Satya Nadella calls AI the "defining technology of our times" and the "ultimate breakthrough" (Marr, 2019).
AI is the attempt to assimilate computer technology with human physiology by formulating computer programs to "make computers smarter". The input of human thinking into machines was a proposition in the 1950's and has been in development ever since. During a conference in the mid1950's a computer and cognitive scientist named John McCarthy, composed the term *"artificial intelligence"* . He is the pioneer who started evolving the field of AI.
As it continues to grow, many industries are seeking for possible opportunities to invest in robotics and development of "thinking" computer systems.

Insurances companies have started using AI from many years within the motor liability insurance segment whilist other lines of business are still conducted in a traditional manner. Nowdays the motor liability insurance process is completely automated: a user requests the insurance coverage, an information system analyses the request, an algortithm valuates the insurance risk and proposes a policy to the user. The *underwrinting phase* where an insurer analyses the business, defines the risks/exposure included and decides to accept or decline the business or defines the premium cost is now an exclusive task take by an automated system.
Recently, some insurance companies have been trying to apply AI in other commercial lines. For example: the insurer company named Elseco created *"ATOM - the insurance Operating System"* an information system that tries to manage all information

about the Space and Aviation Insureance Market. This is a complex line of business since there are a lot of data to analyse whilst most of these data are not digitalized or structured.

Due to this factor it is not possibile to design an AI system as the lack of a data lake does not allow us to manage this type of policy till now like in the case of the motor liability insurance . ATOM is a system focus on creating a data lake but not still on an AI system which evaluates insurance business.

The Insurance of Space and Aviation Risks is going to be the context of this thesis. As a consequence, this thesis will not realize a decision insurance system completely automated but would examine multiple options as to assist the insurer. The thesis will be focused on developing functions to help insurer during the underwrinting phase, as details:

- a system designed to help the insurer comparing a specific business case to similar pre-existing cases;
- a system capable to prevent mistakes by insurer;

The thesis will be presented as follow:

Chapter number 1 describes the present context and the goal of this thesis project. It is introduced the company where the internship has been placed. It is shown the actual management insurance business flow and the management issues/needs highlighted.

Chapter number 2 is an overview of all activities executed during this thesis project. It is shown a Gantt chart and discuss all the phases and activities that have been identified for this project.

Chapter number 3 describes the data analysis phase. The analysis phase is focused to determine the data type and structure of this insurance line. The final output of this phase is the database structure schema.

Chapter number 4 expones the data migration phase with a specific focus on the cleaning and defining data process.

Chapter number 5 describes the business problems that should be resolved. It is also defined the dataset that can be used from AI and probabilistics models.

Chapert number 6 analyses the AI and probabilistic models to solve the business problems. It describes an AI model to clusterize different insurance affairs and a hybrid model that uses AI and probabilistic approch to prevent data entry

error on an information system.

Chapter number 7 describes the process to build the models and use them. It presents all the software's frameworks need as to do the activity, describes all codes produced and analyze all charts useful to evaluate the models.

Chapter number 8 describes the models evaluation process made by insurers.

Chapter number 9 expones the realization of the web application phase. It presents the technology architecture used by the web application.

Chapter 1

Introduce the context

The internship was performed at Oman Insurance Company recently renamed as Sukoon Insurance Company.

Oman Insurance Company P.S.C. ("Sukoon") is a composite insurance company headquartered in Dubai, UAE that sells insurance for individuals and businesses in UAE and Oman[1]. Established in 1975 with majority ownership by Mashreq Bank, Sukoon is one of the largest publicly listed[2] insurers in the far east. Abdul Aziz Abdullah Al Ghurair is the chairman and Jean-Louis Laurent Josi is the CEO of Sukoon.

Figure 1.1: Companies logos

Oman Insurance Company provides Life, Medical, and General insurance covers including products such as Trade Credit insurance. Insurance feature verticals including Healthcare, Motor, Property, Travel, Life, Engineering, Marine Hull, Energy, Marine Cargo, ***Aviation and Space*** and Liability.
The collaboration has been developed both with the IT department and the Aviation & Space commercial business unit.
Aviation & Space commercial business unit is in demand of a new system as to support its activities. The IT department is involved on such task.

1.1 Actual management insurance business flow

1.1.1 Information business flow

On Figure 1.2 is represented a schema wich describes the information business flow in regards to Aviation and Space Insurance Business.

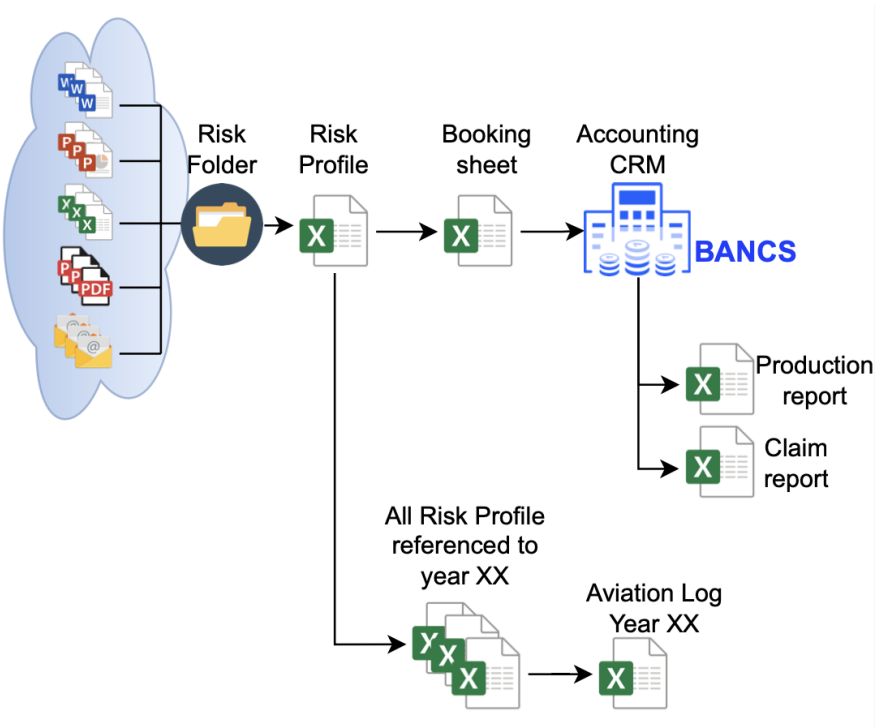

Figure 1.2: Information business flow

The company receives from the clients (represented by the cloud) a set of heterogeneous information: text information (Word documents, Emails), tabular and graphic information (Excel documents, Power Point documents, PDF documents).

Risk Folder is a folder to collect all documents received for a specific insurance risk. These folders contain always different documents and the file structure differs from "Risk Folder" to "Risk Folder".

Risk Profile is an Excel Document that contains the most relevant information about a specific insurance risk. All these "Risk Profile Documents" (one for each insurance risk) have a very similar structure, but not identical.

Aviation Log is an Excel Document providing a table of risks allocated by underwriting year. This document tries to put all information contained in various Risk Profiles in a omogeneous data structure. Aviation Logs (one for each underwriting year) have a very similar structure, but not identical.

Booking Sheet is an Excel Document that summarizes all information necessary to book the insurance risk in the Accounting CRM System called BANCS.

1.1.2 Management business flow

On Figure 1.3 is represented a schema which describes the business flow regarding Aviation and Space Insurance Business. It is represented the full management process from the request of a new insurance coverage by the customer to the policy generation.

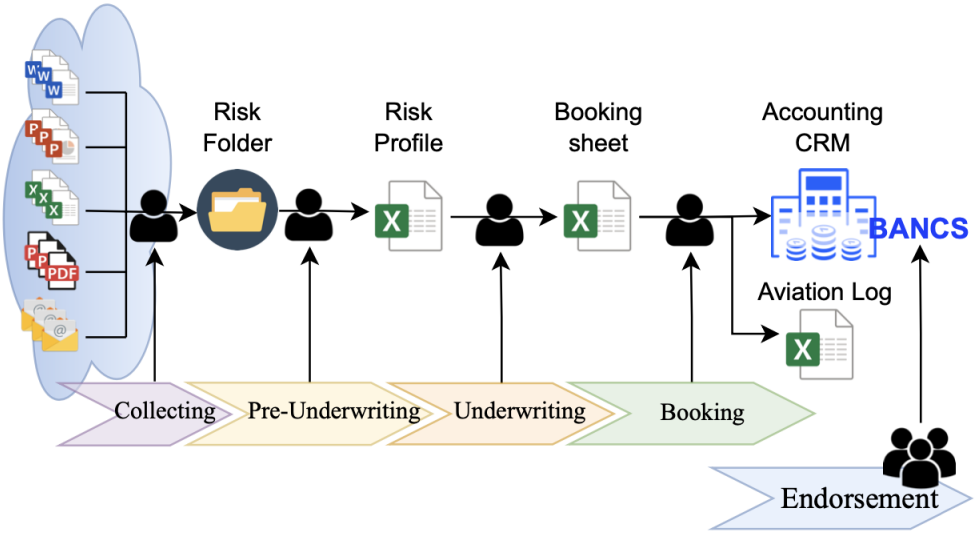

Figure 1.3: Actual management insurance business flow

Starting phase A customer send a request for a new insurance or reinsurance cover to the company. (Oman Insurance provide, for Space and Aviation risks, both insurance and reinsurance services which means the customer can be either a Company or a Broker.)

Collecting phase during this phase a User collects all the files received from Clients whilst archiving on different Risk Folders. The documents pertaining to the same Risk Folder are received by the Insurance company in various batches through the year. All received documents are essential for a correct insurance risk evaluation. This is the reason why such a phase has to be as precise as possible. Any lack of information and/or mis-representation may lead to a foulty risk profile.

Pre-Underwriting phase during this phase a User fills in the Risk Profile by using the information contained in the Risk Folder.

Underwriting phase during this phase a User (the underwriter) fills in the Booking Sheet by using the information contained both in the Risk Profile and the Risk Folder.

Booking phase during this phase a User loads all data contained in the Booking Sheet onto the Accounting CRM called BANCS. During this phase the user records the same information in the underwriting year Aviation Log of reference.

Endorsement phase during this phase a User changes some data in the accounting CRM since specific endorsements may be required in order to redefine certain aspects of the risk profile. This is an optional phase which is not a part of the standard Management business flow.

Ending phase the last phase is correlated to the policy generation. This task is managed with the accounting CRM while the Aviation Space commercial business unit is not accauntable/responsible for this phase of the process anymore .

1.2 Issues and demands management

There are two aspects which influence issues and demands management tasks: Timing and Past knowledge.

Timing The Aviation and Space business is processed in the global market. The vast majority of the insurance risks are renewed/ managed in two periods of the year. During these peak phases, users process incoming business 24hrs/ 24hrs.

This means that a large portion of the book is concentrated in a limited period of time.

Due to the severe work load, the probability of a data entry error becomes very high. Hence, the current business flow looks like as a disadvantage to the efficiency. Furthermore same information being copied and pasted on different documents many times (Risk Profile, Booking Sheet, Aviation Log) increase such a probability even more. Phases highlighted in Figure 1.3 are concentrated in a narrow timeframe.

Past knowledge During the underwriting phase the insurer analyses the risk evaluating all information provided by the customer and reported on Risk profile. During this phase the so called Underwriter decides whether to take a risk, quote the policy premium, pick the percentage of participation (Written line) whilst evaluating some further details. As at now the Underwriter (The Insurer) has no access to the past business profile due to the narrow timeframe (as mentioned, already) and the information data structure does not permit a fast and efficient comparison. Therefore, the Underwriter cannot compare past analyses to the n risk and has to repeat everytime the whole evaluating process for each and every risk. Taking into consideration what has just been described, it looks like that the undeerwriting process has no past knowledge.

1.3 Best case hypothesis: Management business flow

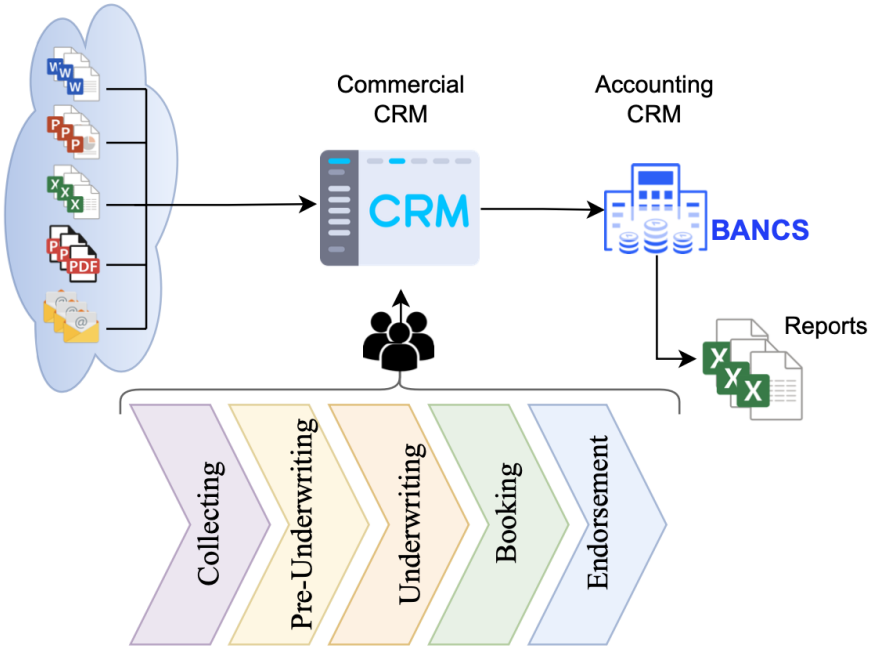

Figure 1.4: Best case hypothesis: Management business flow

On Figure 1.4 is represented an hypothesis to solve/respond to the issues / needs presented in the previous section.
The best case/ hypothesis is a Commercial CRM capable to deal with all management insurance business flow presented before. The Commercial CRM should provide: a database to collect all data, a system to analyse and compare new versus old and/or existing risks and a system to prevent data entry error.

Chapter 2

Gantt chart of all activities

In this chapter there is an overview of all activities executed during this thesis project. On Figure 2.1 it is shown a Gantt chart made at the beginning of the intership as to plan/ organize the activities. All activities are organized by stage and subject. The flags on the chart symbolize the mailstones of the project.

PMI[3] defines : " *A milestone is the planned completion of a significant event in the project. A milestone is not the completion of every task in the project. In an information systems environment, such a milestone might be the completion of the business (macro) design or a successful systems test. In research and development, the approval of the required funding or the completion of a prototype might be considered a milestone.*

...

Milestones are tools used in project management to mark specific points along a project timeline. These points may signal anchors such as a project start and end date, or a need for external review or input and budget checks.

...

In many instances, milestones do not impact project duration. Instead, they focus on major progress points that must be reached to achieve success"

The project is divide in 4 stages :

- Stage 1 : Analysis
- Stage 2 : Data Migration
- Stage 3 : Solving Business Problems
- Stage 4: Realise Web Application

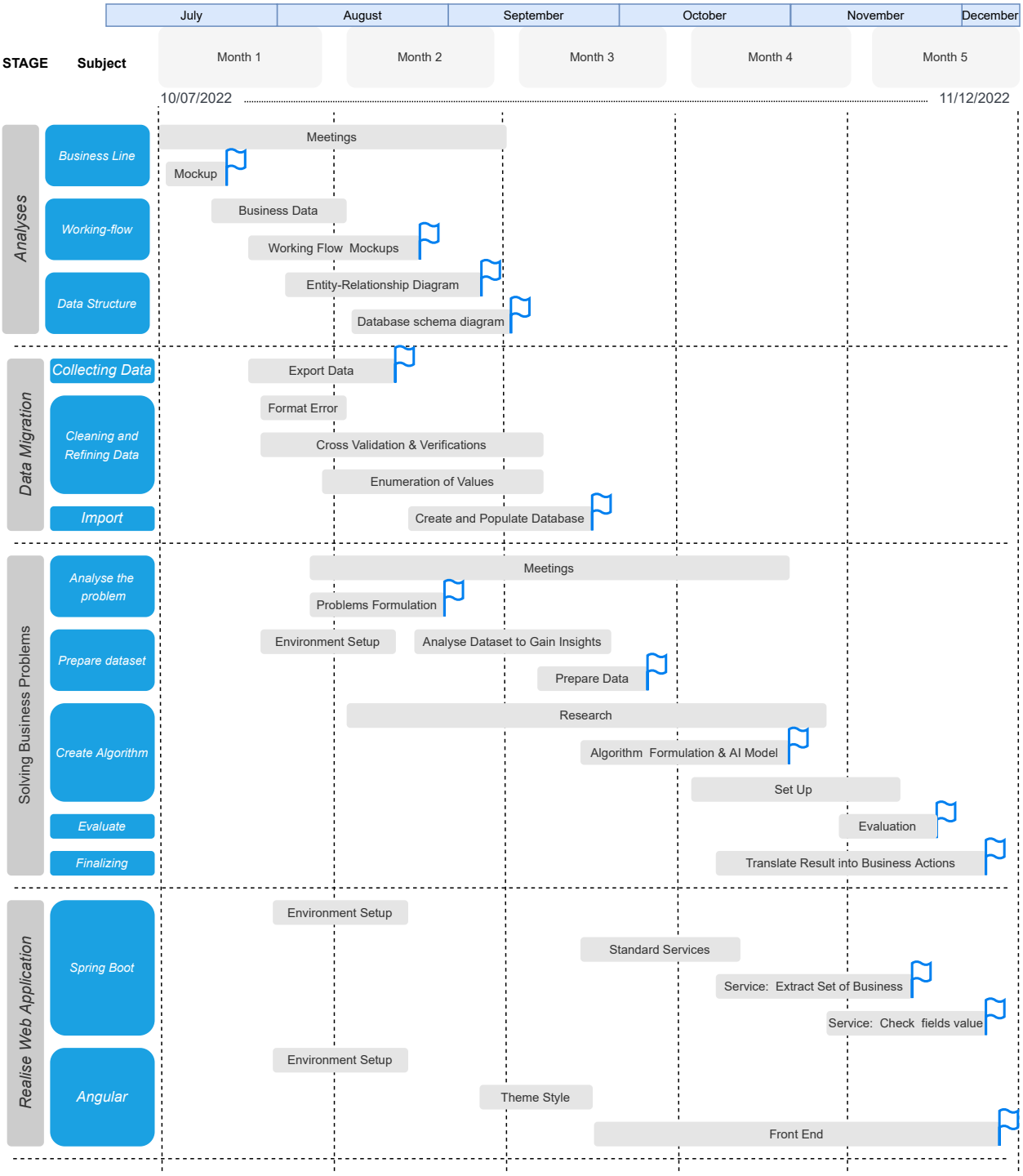

Figure 2.1: Gantt chart of all activities

2.1 Stage 1 - Analysis

At this Analysis stage initial meetings have been arranged in order to plan all activities whilst understanding the insurance aspects of a commercial line with a specific focus on space and aviation. The main target of this phase it is made both by understanding the business and the data at stake (type and structure).
All milestones reached during this stage have been discussed and approved both by the business unit and the IT department.

Subject 1 A - Business Line The milestone of this subject is a mockup that tries to collect all main data treated in a single sheet. The implementation of such a mockup assists both the student and the insurers understanding the data structure.

Subject 1 B - Working-flow The milestone of this subject is a working flow mockup which contributes to get a comprehensive view about the entire data processing during different business phases. This assisted us to merge all different documents arising from various business phases whilst tracing the data flow in a uniformed format.

Subject 1 C - Data Structure The milestones of this subject are both the entity relationship schema and the database schema.

2.2 Stage 2 : Data Migration

In this case, the Data Migration stage is very important because there is no database to manage this part of the business process. In addition, the data are all unstructured on different files and formats.

Subject 2 A - Collecting Data The milestone of this subject is the set of all files needed for the next step. This files set includes: excel and word files produced and managed by the Underwriters alongside with data extractions obtained by an Accounting CRM named BANCS.

Subject 2 B - Cleaning and Refining Data The goal of this subject is to reach a clean and discrete set of data.

Subject 2 C - Import The milestone of this subject is to populate the database designed in the previous stage with clean and refined data.

2.3 Stage 3 : Solving Business Problems

In this Solving Business Problems Stage are considered all meetings arranged to define the business problems, further steps taken to study and analyse an efficient system to resolve the case, additional steps to design/create the system itself.

Subject 3 A - Analyse the problem The goal of this subject is the problems formalization and the definition of the data useful to solving them.

Subject 3 B - Prepare dataset The goal of this subject is to prepare the necessary dataset for the next step.

Subject 3 C - Create Algorithm The goal of this subject is to obtain the models required to resolve the problems.

Subject 3 D - Evaluate The goal of this subject is to select the models made during the previous step which resolve the problems best.

Subject 3 E - Finalizing The goal of this subject is to translate the models realized into business actions.

2.4 Stage 4: Realise Web Application

The goal of this stage is to realize a web-application with a microservices system archietecture that uses the models built in the previous stage.

Subject 4 A - Spring Boot The milestones of this subject are the microservices as to manage the two model designed in the previous stage.

Subject 4 B - Angular The milestone of this subject is the frontend part of the web-application that utilizes the two microservices realized.

Chapter 3

Data Analysis phase

This chapter describes the analysis phase which is focused to determine the data type and data structure of this insurance line. The final output of this phase is the database structure schema.

3.1 Mock-Up

From Cambridge Business English Dictionary : "*Mock-Up is a model of something, which shows how it will look or operate when it is built, or which is used when the real thing is not yet available*"[4]

Applied to IT : "*A mockup is a conceptual tool that is used especially in web development. It is basically an early draft of a website or web application. Mockups are primarily used for conception to convert ideas and concepts into a concrete design. They typically include the final navigation structure and detailed design elements so that they often resemble the final design of the website.*"[5]

The mock-ups presented on this chapert are not properly an application mock-up but they provide a rappresentation of the data at stake and the structure of them on "a piece of paper".
This type of mock-ups is one of the best tools that can be used during the analysis sessions between users and analysts. It helps both users to describe their business and analysts to make question usefull to understand tha data structure.

On Figure 3.1 is represented the last version of the Mock-up obtain after a series of calls conference between the Business Unit and the student. This mock-up show all the main data involved on this type of business and have an hight level of personalization. It allows to the user to manage different type of business like Airplanes insurances, Airports Insurances or Satellites Insurances with an homogeneous data structure.

The Mock-up on Figure 3.1 is divided in nine sections defined as follow:

Section 1 - Business Details This section represents all main information relative to the type of insurance risk.

Section2 - Policy Details This section represents all main information correlated to a certain insurance policy

Section 3 - Client This section shows the list of the clients defined as "The Insureds". Every insurance policy has one insured, at least.

Section 4 - Provider This section represents the list of business providers. However a direct insurance policy has never a provider whilst a reinsurance/ indirect policy has one provider, at least. A Provider is named as Broker of risks. The normally represents international organization specialized in trading policies.

Section 5 - Leader This section has data, only if the risk corresponds to a reinsurance/ indirect business. Any other reinsurance company defined as "the Leader" is herewith named. This Company provides the market with the quote/ price related to the same policy; whilst a percentage that corresponds to the level of participation on this risk is defined.

Section 6 - Risk domicile This section identifies the domiciles of a certain risk/policy.

Section 7 - Type of Risk This is a sub-segment of the section "Risk domicile" as for each and every Risk domicile we are entitled to define a list of types of risk. Every sub-section called "type of risk" represents the list of different interests/ risks included within a policy for a specific risk domicile. On this mock-up there is only the main data related to a type of risk. The insurer can definep in the same policy various types of risk.

Section 8 - Acquisition Costs It is a section that allows the insurer to account diverse type of cost related to a certain policy.

Section 9 - Summary This section summarizes all data that can be determined from other data included in this mockup. It is a very useful technical overview for the Insurer during the underwriting phase.

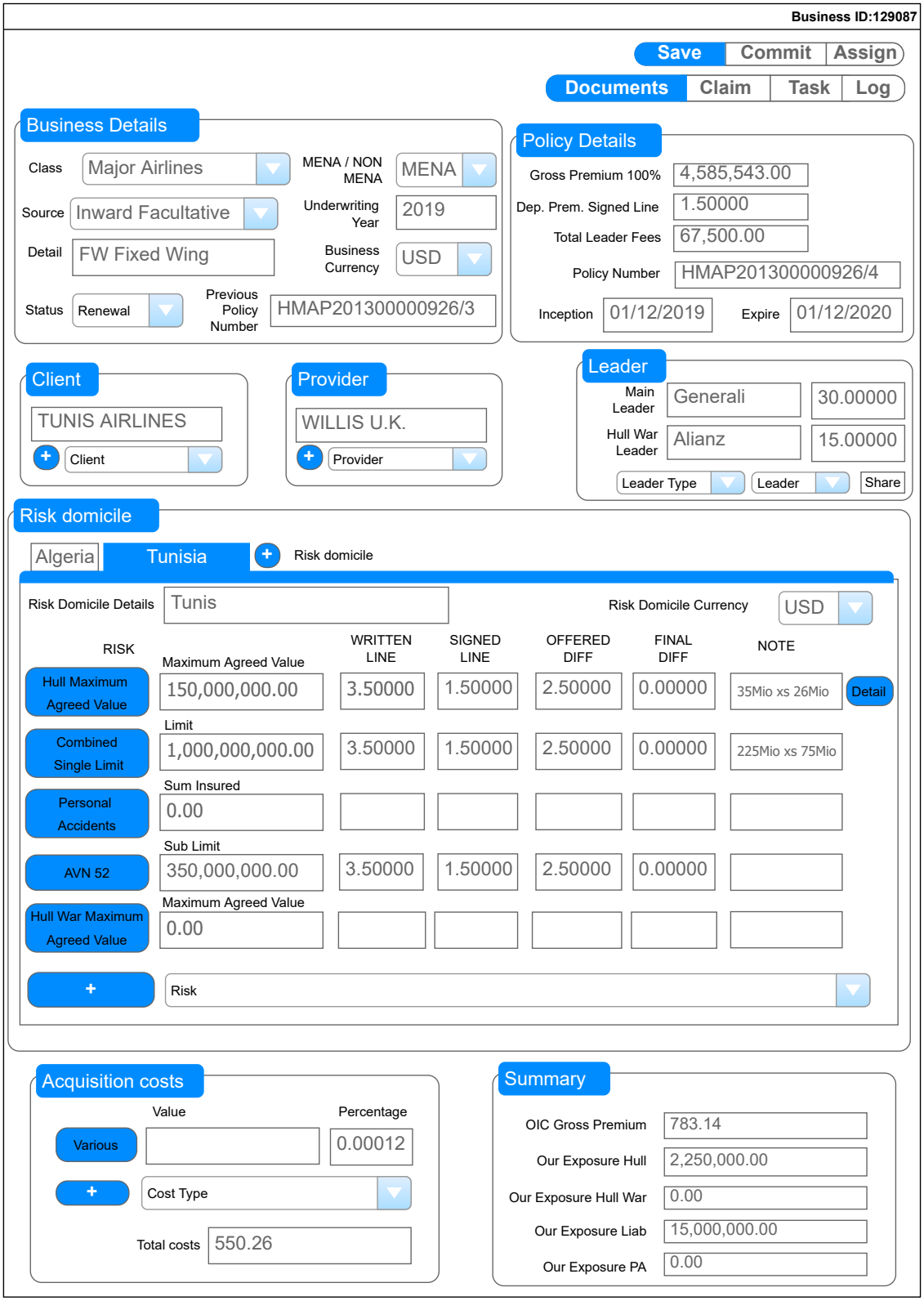

Figure 3.1: MockUp (last revision)

3.1.1 Calculation algorithm

There are two ways of calculating data included in the section named as "Summary" which depend upon the data correlated to the business itself.

NOTE: The green data fields are used to calculate the yellow data fields, the "Di" represents the i-th risk domicile and the "Ei" represents the i-th acquisition costs.

Figure 3.2: MockUp: Calculation algorithm of case A

The first calculation algorithm (case A) is represented on Figure 3.2 once the policy has a unique Deposit Premium Signed Line (see in the "Policy details" section). All past business managed by the insurance company have the same structure.

The second calculation alorithm (case B) is represented on Figure 3.3 once the policy does not have a unique Deposit Premium Signed Line, but a Premium Allocation has made for each and every type of risk (see the "Type of Risk" section).

Figure 3.3: MockUp: Calculation algorithm of case B

3.2 Business Phases Mock-Ups

In the previous section is analyzed the data structure whilst here it is defined which data are involved on each and every of the business phases mentioned in the Chapter number 1.

The last revision of Mock-up is represented on Figure 3.4 with the Business Phase section on the top.

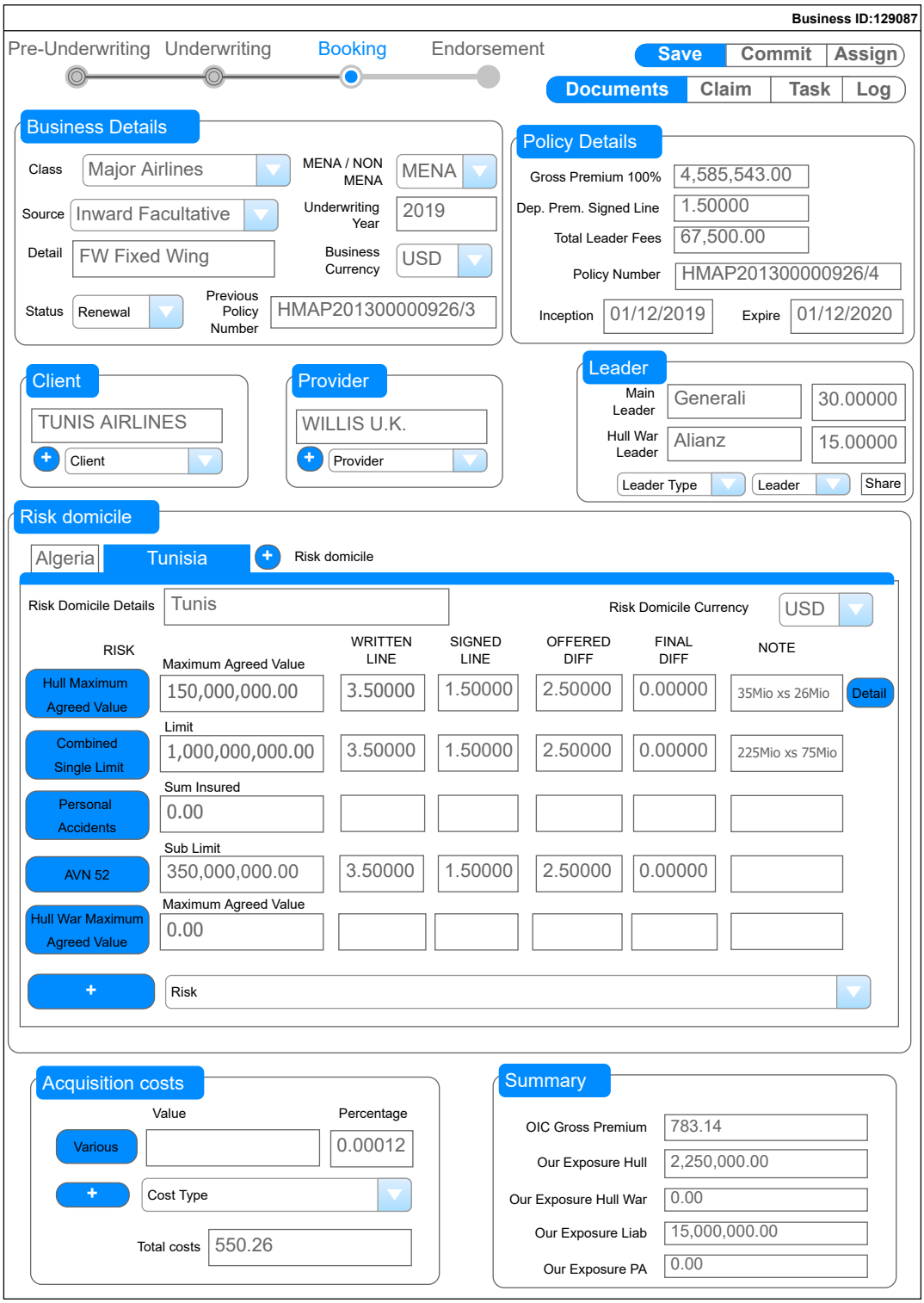

Figure 3.4: MockUp (last revision) with Business phases section

The data fields high-lighted in the Picture 3.5 are a set of data that a junior underwriter fills once it is preparing the "Risk profile" (see Chapter number 1). All these data represent the basic information required during the underwriting phase.

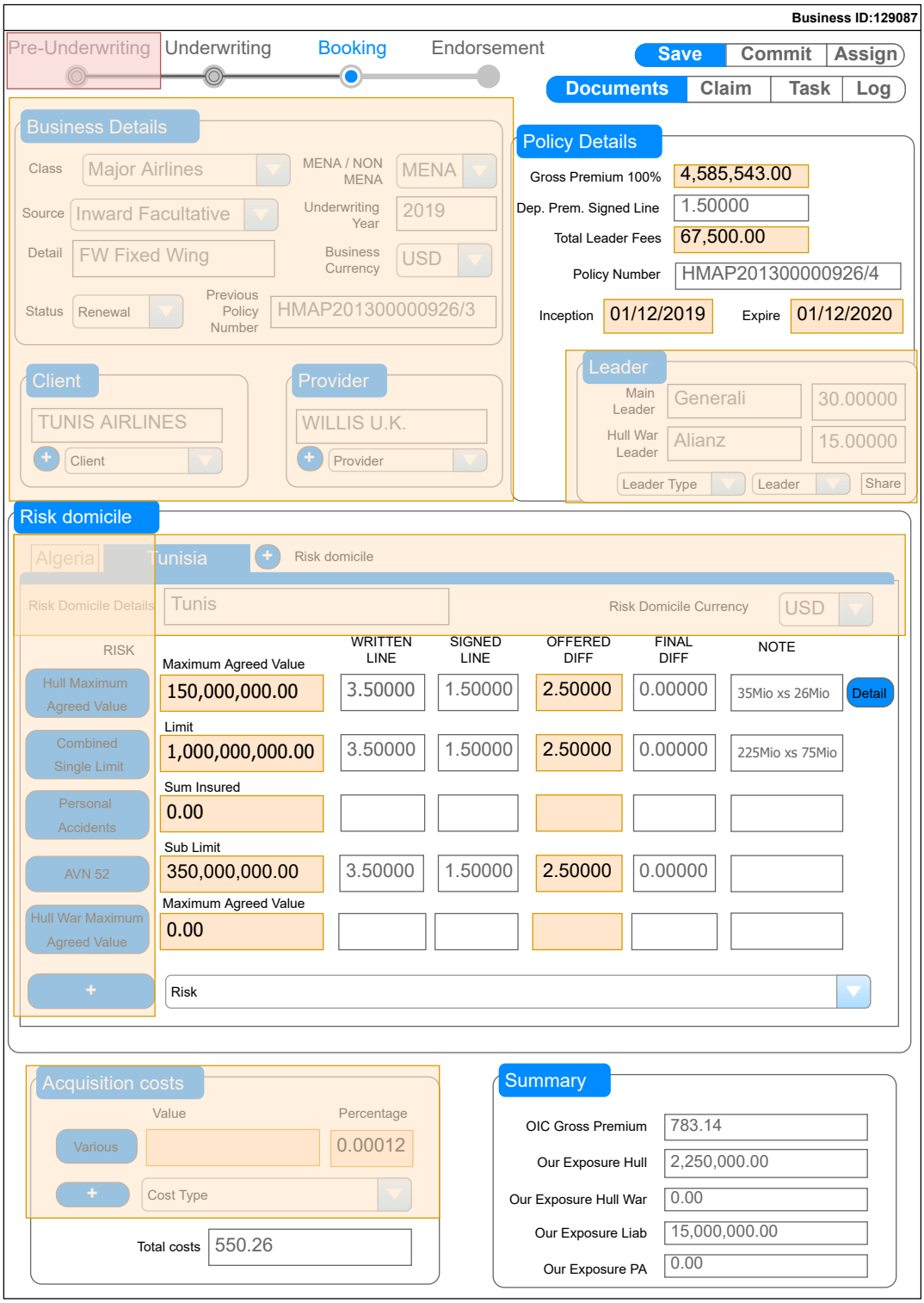

Figure 3.5: MockUp: Pre-Underwriting (Phase 1 of 4)

The data fields high-lighted in the Picture 3.6 are a set of data that insurer fills in once the underwriting analysis phase is finalized.
These data are the core of each and every business risk. These figures are the result of the commercial trade happening between the Providers (Brokers) and the Insurers.

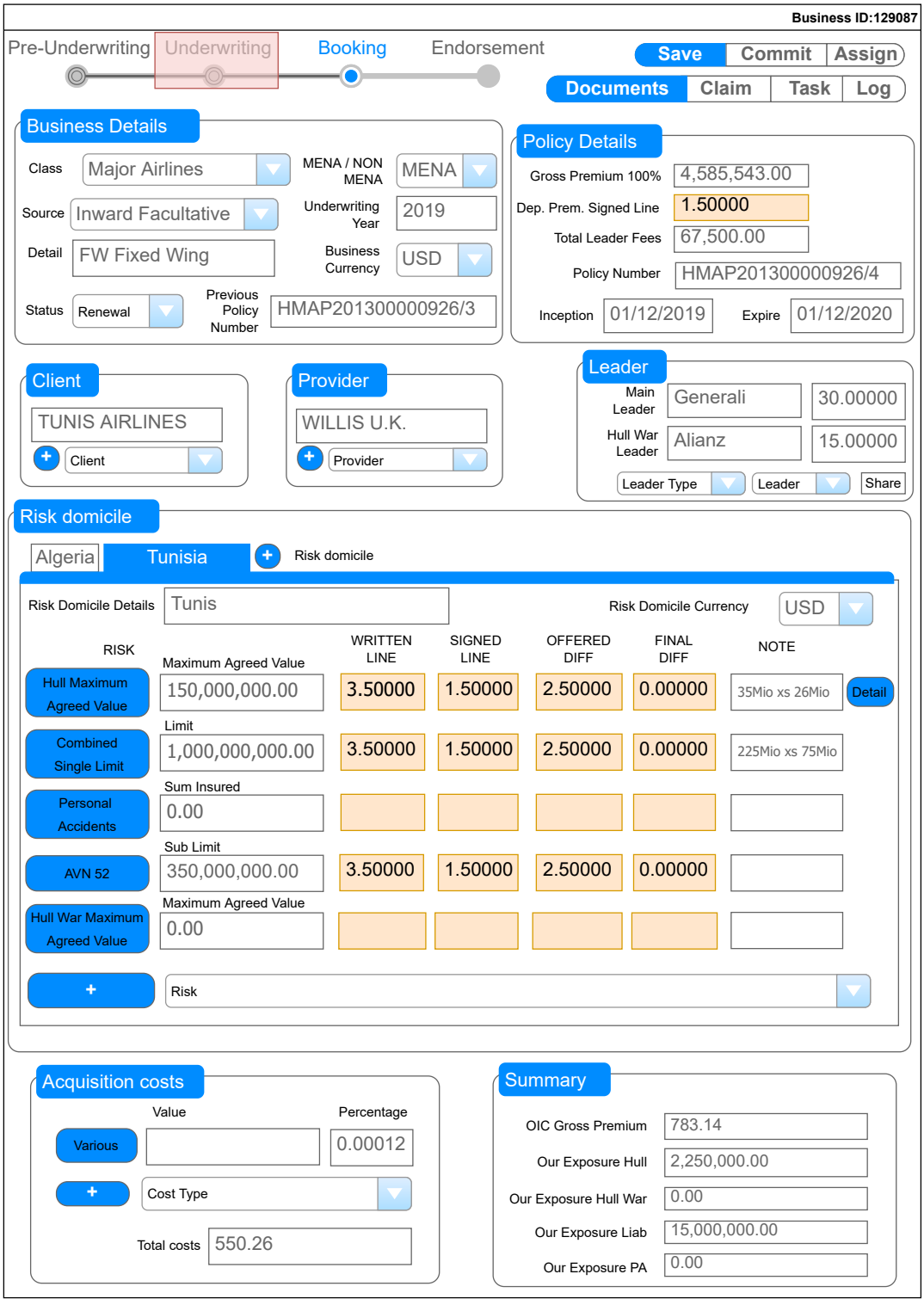

Figure 3.6: Mock-Up: Underwriting (Phase 2 of 4)

Once a policy is booked in the accounting CRM, then the system reflects a policy number. For such a reason, the only data filled and/ or modified during the booking phase is the Policy number (see Figure number 3.7).

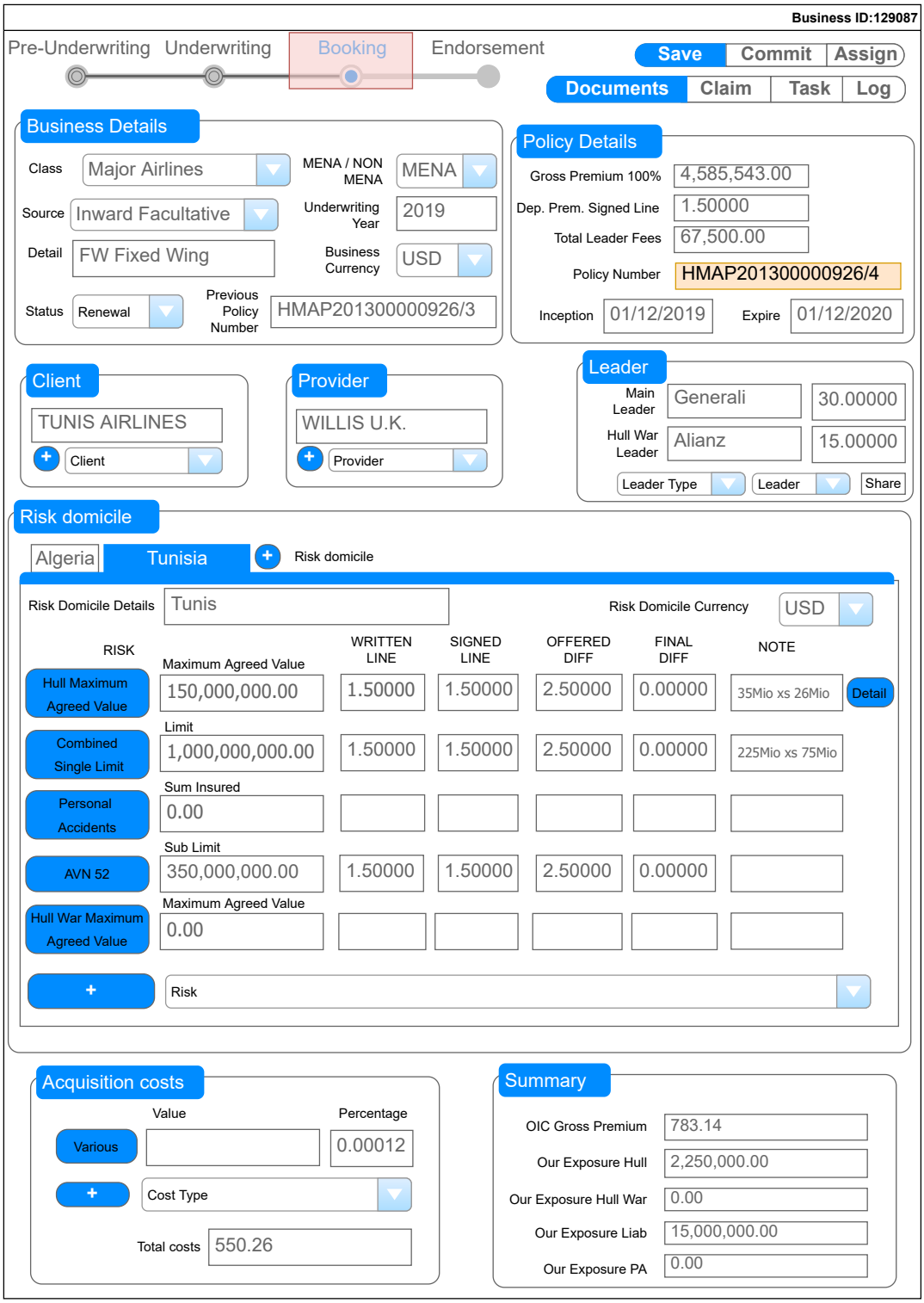

Figure 3.7: MockUp: Booking (Phase 3 of 4)

The data fields high-lighted in the Picture 3.8 are a set of data that can be modified/ altered after the policy emission. These alterations phases named as "Endorsements" can be executed various times during he course of a policy period once/ if required.

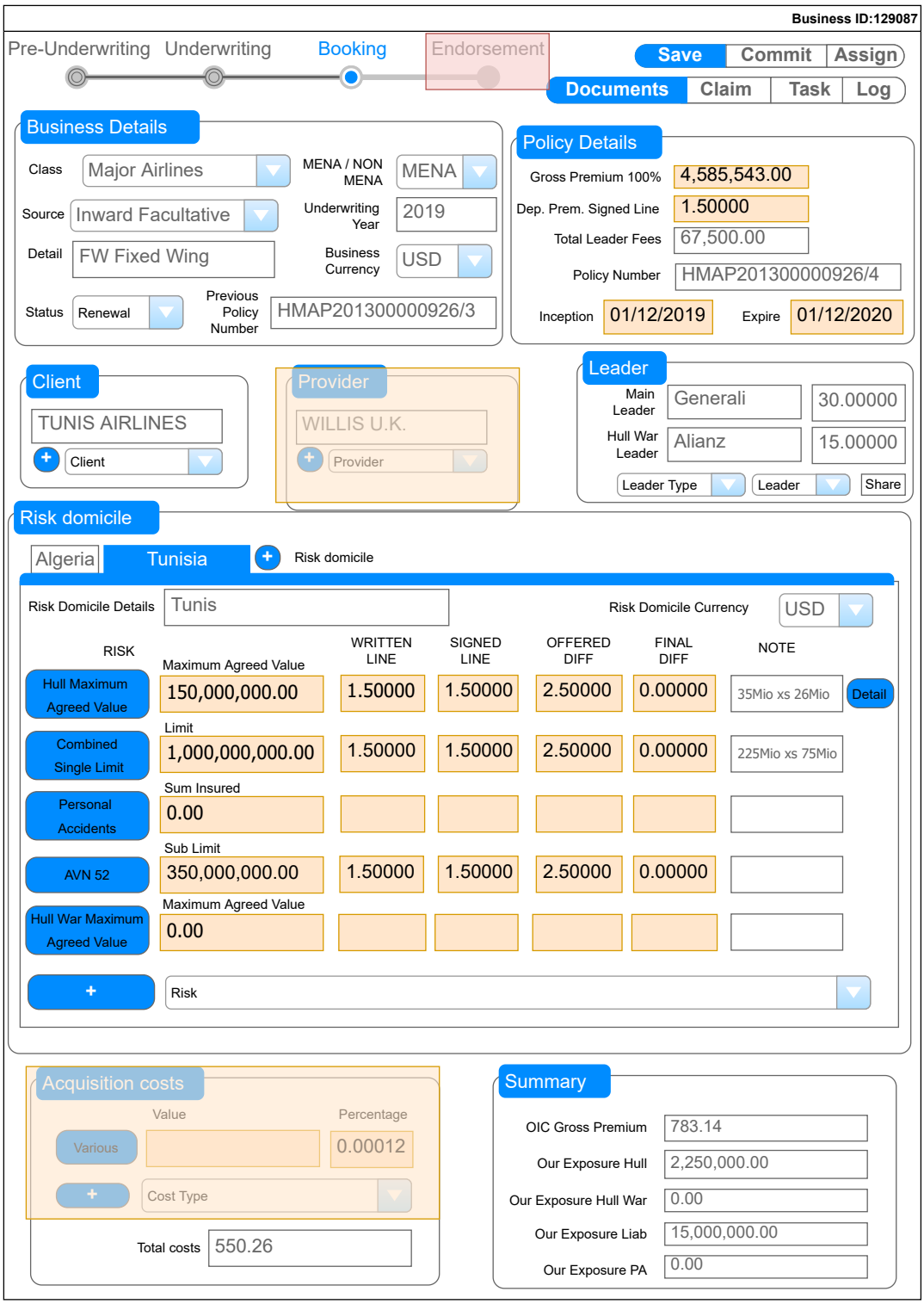

Figure 3.8: MockUp: Endorsement (Phase 4 of 4)

3.3 Entity Relationship Diagram

Starting from the mock-ups shown in the previous section and after an another set of meetings it has been obtained the entity/relationship diagram shown in figure 3.9.

3.4 Data dictionary

The Entities represented in Figure 3.9 are described in Table 3.1.

The Relationships represented in Figure 3.9 are described in Table 3.2.

The Attributes related to Entities or Relationships represented in Figure 3.9 are described in Table 3.3.

Table 3.1: Entities

Entity	Description	Attributes	identifier
BUSINESS	This entity describes a single insurance business.	ID BUSINESS, Business Status, MENA-NON MENA, Detail, underwriting Year, Status	ID BUSINESS
BUSINESS PHASE	This entity describes a Phase of a single insurance business.	Phase Status, Phase Start Date, Phase End date, Total Leader Fees, Gross Premium 100, Inception policy Date, Expiry Policy Date, Flag Premium Allocation, Deposit Premium Signed Line	COD PHASE, ID BUSINESS
CLIENT	This entity describes a client of the insurance company.	ID CLIENT, Name	ID CLIENT
COST	This entity describes an acquisition cost related to an insurance business.	Value, Percentage	COD PHASE, ID BUSINESS, ID COST TYPE
COST TYPE	This entity describes a type of acquisition cost.	ID COST TYPE, Name	ID COST TYPE
COUNTRY	This entity describes a country.	ID COUNTRY,Name, Region	ID COUNTRY
CLAIM	This entity describes an insurance claim related to an insurance business.	CLAIM NUMBER, data of loss, Loss description, Claim year, Claimed Amount, Payment amount	CLAIM NUMBER
CLASS	This entity describes a type of insurance business class.	ID CLASS, Name	ID CLASS
CURRENCY	This entity describes a currency.	CURRENCY CODE, Name, Exchange Rate	CURRENCY CODE

Continued on next page

Table 3.1 – continued from previous page

Entity	Description	Attributes	identifier
DOCUMENT	This entity describes a document related to an insurance business.	DT INSERT, File, Name File	COD PHASE, ID BUSINESS, ID DOCUMENT TYPE, DT INSERT
DOCUMENT TYPE	This entity describes a document type.	ID DOCUMENT TYPE, Name	ID DOCUMENT TYPE
EXPOSURE LEVEL	This entity describes the risk exposure levels related to an insurance business.	ID EXPOSURE LEVEL, Exposure Level	ID EXPOSURE LEVEL
EXPOSURE TYPE	This entity describes the exposure type of risk related to an insurance business.	COD EXPOSURE TYPE, Name	COD EXPOSURE TYPE
INSURER COMPANY	This entity describes another insurance company involved in the same insurance business.	ID INSURER COMPANY, Name	ID INSURER COMPANY
LEADER TYPE	This entity describes the leader's type (Main Leader, Hull War Leader, etc...)	ID LEADER TYPE, Name	ID LEADER TYPE
NOTE	This entity describes a note related to an insurance business phase .	DT INSERT, Text	COD PHASE, ID BUSINESS, DT INSERT
PHASE	This entity describes a Phase of an insurance business activity.	COD PHASE, Name	COD PHASE
POLICY	This entity describes the Policy generated and related to an insurance business.	POLICY NUMBER	POLICY NUMBER
PROVIDER	This entity describes the provider of an Insurance business, IE The Insurance broker.	ID PROVIDER, Name	ID PROVIDER
RISK DOMICILE	This entity describes a risk domicile of an insurance business activity.	Detail	COD PHASE, ID BUSINESS, ID COUNTRY, COD RISK TYPE
RISK TYPE	This entity describes a type of risk related to an insurance business.	COD RISK TYPE, Name, Exposure Label	COD RISK TYPE
SOURCE	This entity describes a type of insurance business source.	ID SOURCE, Name	ID SOURCE

Table 3.2

Relationship	Description	Components	Attributes
Attached 1	It means a note related to a business phase.	NOTE, BUSINESS PHASE	-
Attached 2	It means a document related to a business phase.	DOCUMENTO, BUSINESS PHASE	-
Attached 3	It means the source of an insurance business.	SOURCE, BUSINESS	-

Continued on next page

Table 3.2 – continued from previous page

Relationship	Description	Components	Attributes
Attached 4	It means the class type of an insurance business.	CLASS, BUSINESS	-
Broker	It means the broker of an insurance business.	PROVIDER, BUSINESS PHASE	-
Business Currency	It means the currency of an insurance business.	CURRENCY, BUSINESS	-
Claim Currency	It means the currency of an insurance claim.	CURRENCY, CLAIM	-
Comprise	It means that an insurance business comprises various costs.	COST, BUSINESS PHASE	-
Country Outlook	It means an Outlook (type and exposure of an insurance level of risk) related to a Country	COUNTRY, EXPOSURE TYPE, EXPOSURE LEVEL	-
Current Policy	It means an insurance policy related to an insurance business.	POLICY, BUSINESS PHASE	-
Domiciled	It means an insurance risk domiciled in a country.	COUNTRY, RISK DOMICILE	-
Insured	It means a client related to a business phase.	CLIENT	-
Interest	It means a specific risk which is referred to an insurance business.	RISK TYPE, RISK DOMICILE	Exposure, Deposit Premium Allocation, Note, Signed Line, Written Line, Offered Differential, Final Differential
Leader	It means the leader of an insurance business.	INSURER COMPANY, LEADER TYPE, BUSINESS PHASE	Share quota
Previous Policy	It means a previous insurance policy related to the current insurance policy	POLICY, BUSINESS	-
Refer To (1)	It means a business phase related to a business.	BUSINESS, BUSINESS PHASE	-
Refer To (2)	It means that a cost is associated to a typology of cost.	COST, COST TYPE	-
Refer To (3)	It means a type of phase related to a business phase.	PHASE, BUSINESS PHASE	-
Refer To (4)	It means a type of document related to a document.	DOCUMENT, DOCUMENT TYPE	-
Refer To (5)	It means an insurance claim related to a policy.	CLAIM, POLICY	-
Refer To (6)	It means a risk domicile related to a business phase.	RISK DOMICILE, BUSINESS PHASE	-
Risk Domicile Currency	It means the currency of an insurance risk domicile.	RISK DOMICILE, CURRENCY	-
Risk Exposure	It is the exposure to a specific risk type.	RISK TYPE, EXPOSURE TYPE	-

Table 3.3: Attributes

Attribute	Entity/Relationship	Domain	Description
ID BUSINESS	BUSINESS	Integer (sequence)	Unique identification code to identify a specific Insurance Business
Business Status	BUSINESS	String	This attribute represents the status of a specific insurance business [C: Close, O: Open]
MENA / NON MENA	BUSINESS	String	The representation of the domicile of a specific insurance business (MENA means Middle East and North Africa) [N: NON MENA, M: MENA]
Detail	BUSINESS	String (with enumeration)	This is a detail of a specific insurance business
Underwriting year	BUSINESS	Integer	This attribute represents the underwriting year of a specific insurance business
Status	BUSINESS	String (with enumeration)	This attribute represents the status of a specific insurance business. [R: Renewal Policy, N: New Policy]
Phase Status	BUSINESS PHASE	String (with enumeration)	This attribute represents the phase of a specific insurance business is represented herewith.
End phase date	BUSINESS PHASE	Date	This attribute represents the ending date of a specific phase of insurance business.
Start phase Date	BUSINESS PHASE	Date	This attribute represents the starting date of a specific phase of insurance business.
Total Leader Fees	BUSINESS PHASE	Float	This attribute represents the Total Leader Fees for a specific insurance business is shown herewith.
Gross Premium 100	BUSINESS PHASE	Float	This attribute represents the Premium Policy 100% for a specific insurance business.
Flag Premium Allocation	BUSINESS PHASE	String (with enumeration)	This is a configuration flag for the specific insurance business. [T: Presence of Premium Allocation, F: Absence of Premium Allocation]
Deposit Premium Signed Line	BUSINESS PHASE	Float	This attribute represents the Deposit Premium Signed Line valorized in case of Flag Premium Allocation ='F'.
Expiry policy	BUSINESS PHASE	Date	This attribute represents the Expiry policy Date of a specific insurance business.
Inception Policy	BUSINESS PHASE	Date	This attribute represents the Inception policy Date of a specific insurance business.
ID CLIENT	CLIENT	Integer (sequence)	This is the Unique identification code to identify the Client entity.

Continued on next page

Table 3.3 – continued from previous page

Attribute	Entity/ Relationship	Domain	Description
Name	CLIENT	String	This attribute represents the name of the client.
Value	COST	Float	This attribute represents the Value of a cost.
Percentage	COST	Float	This attribute represents the cost percentage referred to the the gross premium 100.
ID COST TYPE	COST TYPE	Integer (sequence)	This is the Unique identification code to identify the Cost Type entity.
Name	COST TYPE	String	This attribute represents the cost type's name.
ID COUNTRY	COUNTRY	Integer (sequence)	This is the Unique identification code to identify the Country.
Name	COUNTRY	String	This attribute represents the Name of the Country.
Region	COUNTRY	String (with enumeration)	This attribute represents the Region of the Country. [Values: Australasia, South Asia, Western Europe, Carribean, Central America, Africa, Middle East, South Africa, Fas East Asia, Eastern Europe, North Africa, Central Asia, North America]
CLAIM NUMBER	CLAIM	String	This attribute represents the Claim's number. It is a Unique identification generated by tha accounting CRM.
Data of loss	CLAIM	Date	This attribute represents the data related to a Claim.
Loss description	CLAIM	String	This attribute represents the description related to a Claim.
Claim Year	CLAIM	Integer	This attribute represents the year related to a Claim.
Claim Amount	CLAIM	Float	This attribute represents the Amount related to a Claim.
Payment Amount	CLAIM	Float	This attribute represents the Payment Amount related to a Claim.
ID CLASS	CLASS	Integer (sequence)	This is the Unique identification code to identify the Class entity.
Name	CLASS	String	This represents the Class' name.
Currency Code	CURRENCY	String	This is the Unique identification code to identify the Currency entity.
Name	CURRENCY	String	This attribute represents the Name of the Currency.
Exchange rate	CURRENCY	Float	This attribute represents the Currency's Exchange rate.
DT INSERT	DOCUMENT	Datetime	This attribute represents the Insert Date of a Document related to an insurance business phase

Continued on next page

Table 3.3 – continued from previous page

Attribute	Entity/ Relationship	Domain	Description
File	DOCUMENT	File	This attribute represents the binary file related to a document
Name File	DOCUMENT	String	This attribute represents the name of the file related to a document
ID DOCUMENT TYPE	DOCUMENT TYPE	Integer (sequence)	This is the Unique identification code as to identify the Document Type entity.
Name	DOCUMENT TYPE	String	This attribute represents the Name of the Document Type.
ID EXPOSURE LEVEL	EXPOSURE LEVEL	Integer (sequence)	This is Unique identification code to identify the Exposure Level entity.
Exposure Level	EXPOSURE LEVEL	Float (with enumeration)	This attribute represents the Exposure Level value. [0, 20, 40, 60, 80, 100]
COD EXPOSURE TYPE	EXPOSURE TYPE	String	This is the Unique identification code to identify the Exposure Type entity
Name	EXPOSURE TYPE	String	This attribute represents the Exposure Type's name.
ID INSURER COMPANY	INSURER COMPANY	Integer (sequence)	This is the Unique identification code to identify the Insurer entity.
Name	INSURER COMPANY	String	This attribute represents the Name of the Insurance Company.
ID LEADER TYPE	LEADER TYPE	Integer (sequence)	This is the Unique identification code to identify the Leader's Type entity.
Name	LEADER TYPE	String	This attribute represents the Leader's Type name.
DT INSERT	NOTE	Date	This attribute represents the Insert Date of a Note related to an insurance business phase
Text	NOTE	String	This attribute represents a Note related to an insurance business phase.
COD PHASE	PHASE	String	This is the Unique identification code to identify the Phase entity
Name	PHASE	String	This attribute represents the Name of the Business Phase.
Policy Number	POLICY	String	This attribute represents the Policy Number. It is a Unique identification generated by tha accounting CRM.
ID PROVIDER	PROVIDER	Integer (sequence)	Unique identification code to identify the Provider entity.
Name	PROVIDER	String	This attribute represents the Name of the Provider.
Detail	RISK DOMICILE	String	This attribute represents a Detail related to an insurance risk domicile.
ID RISK TYPE	RISK TYPE	Integer (sequence)	This is the Unique identification code to identify the Risk Type entity.

Continued on next page

Table 3.3 – continued from previous page

Attribute	Entity/ Relationship	Domain	Description
Name	RISK TYPE	String	This attribute represents the Name of the Risk Type.
Exposure Label	RISK TYPE	String	This attribute represents the Label of the Risk Type.
ID SOURCE	SOURCE	Integer (sequence)	This is the Unique identification code to identify the Source entity.
Name	SOURCE	String	This attribute represents the Name of the Source.
Exposure	Interest	Float	This attribute represents the Exposure related to an insurance business interest for a specific risk domicile.
Deposit Premium Allocation	Interest	Float	This attribute represents the Deposit Premium Allocation related to an insurance business interest for a specific risk domicile.
Note	Interest	String	This attribute represents a Note related to an insurance business interest for a specific risk domicile.
Signed Line	Interest	Float	This attribute represents a Signed Line related to an insurance business interest for a specific risk domicile.
Written Line	Interest	Float	This attribute represents a Written Line related to an insurance business interest for a specific risk domicile.
Offered Differential	Interest	Float	This attribute represents a Offered Differential related to an insurance business interest for a specific risk domicile.
Final Differential	Interest	Float	This attribute represents a Final Differential related to an insurance business interest for a specific risk domicile.
Share quota	Leader	Float	This attribute represents a Share related to a leader for a specific insurance business.

3.5 Data Base Schema

The ER Model is intended as a description of real-world entities. Although it is constructed in such a way as to allow easy translation to the relational schema model, this is not an entirely trivial process. The ER diagram (Figure 3.9) represents the conceptual level of database design meanwhile the relational schema (Figure 3.10) is the logical level for the database design.
Tables in the database management system (DBMS) are represented in Figure 3.10.

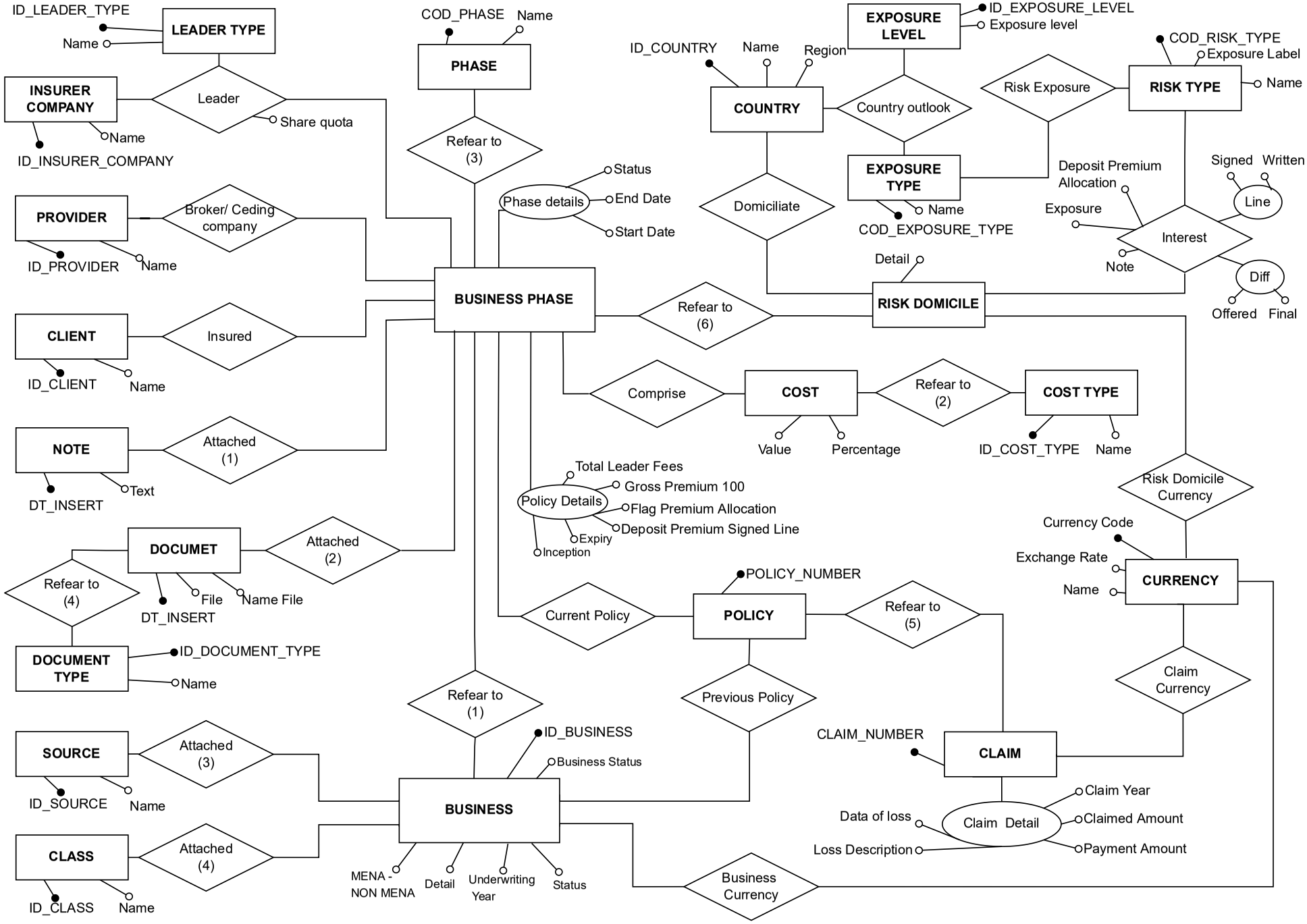

Figure 3.9: Entity/Relationship Diagram

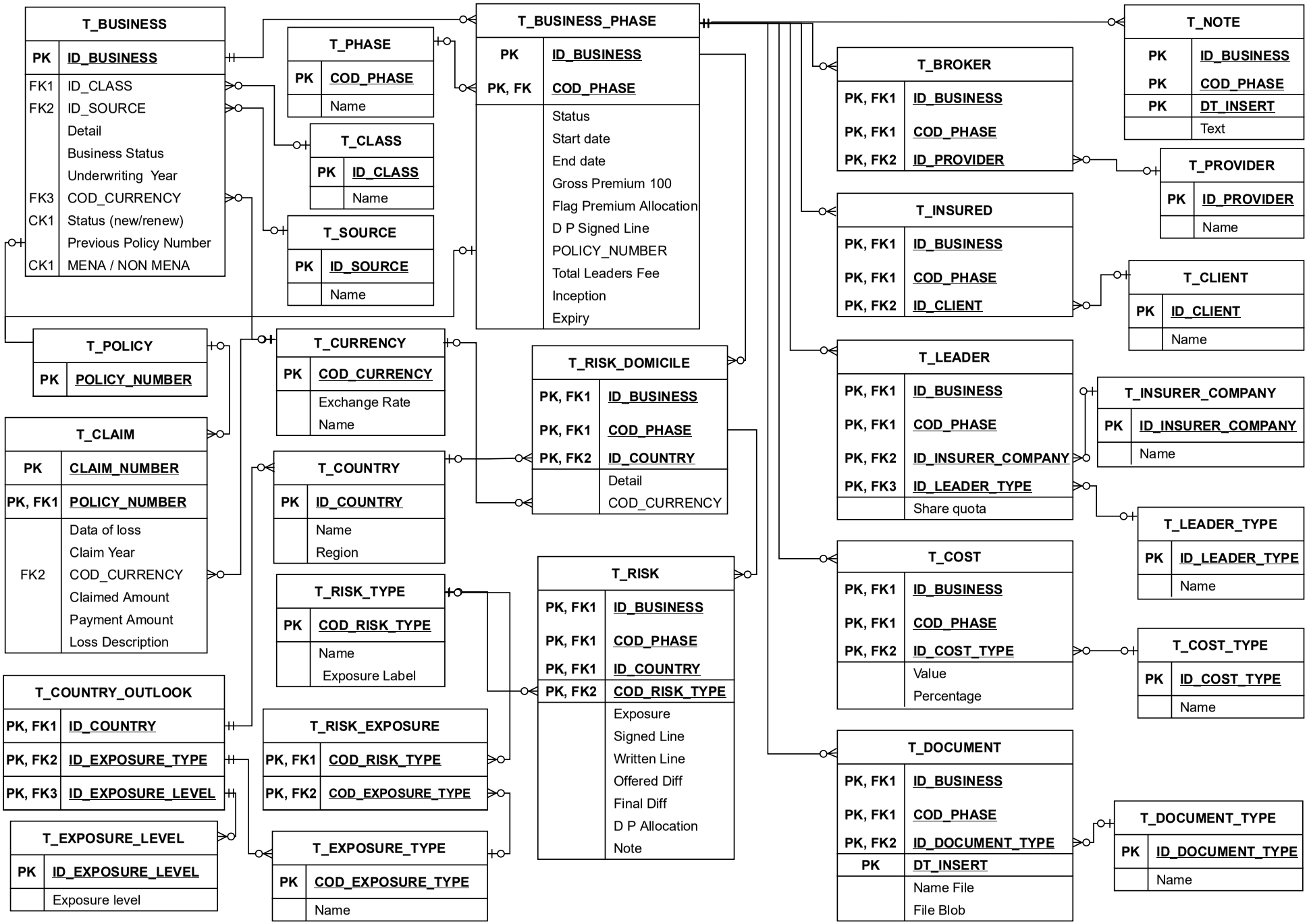

Figure 3.10: Database schema

Chapter 4

Data migration phase

On this chapter is exposed the data migration phase with a specific focus on the cleaning and defining data process. This phase is massive important to obtain good results on the next phases of the project.

4.1 Environment preparation

Before starting the "Data migration phase" is necessary to prepare the environment needed for these phase: a database to fullfill the whole data.
Because this company has already an Oracle database and Oracle databases provide also a lot of utilities to clean data, the data are migrated on this platform.

The student needs an exclusive local database to make the thesis work easier like the Oracle Database Express Edition[1].
This thesis work is made by using an Apple Macbook Pro M1 notebook, but unfortunately Oracle databases are not compatible with the new Apple Silicon CPU architecture M1. Due to this factor, it is not possible to run an Oracle XE database on this notebook type.
To overcome this issue it is needed to run the Oracle XE database on an x86_64 virtual machine and the easiest way to get this done it is to use Colima: a tool to have container runtimes on macOS with minimal setup.

The steps to prepare the desired environment are:

1. Installing colima [7];
2. Downloading an Oracle XE docker image [8];

[1] Oracle Database Express Edition (XE) is the same powerful Oracle Database that enterprises rely on Worldwide, packaged for simple download, ease-of-use, and a full-featured experience. [6]

3. Running colima
   `run colima start --arch x86\_64 --memory 4`
   The `--arch x64\_86` is the important part to set up the CPU architecture as to be compatible with the Oracle XE docker images. The additional memory-flag is for performance reasons;

4. Creating and Running a Docker with Oracle XE container
   `docker run -d -p 1521:1521 -e ORACLE_PASSWORD=Pass123  gvenzl/oracle-xe`

5. Downloading and Installing SQL Developer[2] [9];

6. Create the Database for the project following the database schema introduced on Chapter 3 - Figure 3.10.
   The database schema generated by the SQL Developer application after the database creation is presented on Figure. 4.1;

4.2 Collecting, Cleaning and defining data

The first step to migrate on the database all data saved on different files and formats is to collect all the possible data that are currently managed by the business unit. These documents/data have been collected during the "Collecting, Cleaning and defining data" process:

Aviation Logs The last 10 Aviation Log Files have been collected (starting from year 2013 to year 2022). As described on Chapter 1, the aviation logs have a similar, but not identical data structure and format. The Data Migration phase uses these files as the main source of data. Others documents/data collected were used to refining, cleaning and check this data, although.

Policy Reports From the accounting CRM named BANCS it has been extracted a report containing all Aviation and Space Insurance Policies related to the last 10 years written/subscribed/managed by the Insurer. This extraction is split in two sub reports since during the course of the years the company changed the accounting CRM system. This extraction contains both the policy number and the main data regarding a specific policy as : Insured name, premium amount, policy data inception, policy data expiry, signed line. The extraction was used during the data cross-check on "Aviation Log" data both to discorver

[2]Oracle SQL Developer is a free, integrated development environment that simplifies the development and management of Oracle Database in both traditional and Cloud deployments. SQL Developer offers complete end-to-end development of your PL/SQL applications, a worksheet for running queries and scripts, a DBA console for managing the database, a reports interface, a complete data modeling solution, and a migration platform for moving your 3rd party databases to Oracle.

and manage data error or inconsistencies.

Claim Reports From the accounting CRM named BANCS it has been extracted a report containing all Aviation and Space Claims of the last 10 years managed by the company. This extraction is split in two sub report since during the course of the years the company changed the accounting CRM system. This extraction contains both the claim number and the main data regard to the claim as : Loss description, Claim Amount, Payment amount, Transaction Date, relevant Policy Number. The extraction was used during the data cross-check on "Aviation Log" data to discorver and manage data error or inconsistencies.

Risk Profile These data files were mainly used to understand the business during the analysis phase. During this phase these data were extremely useful as to check some data inconsistencies related to a specific Insurance Risk emerged from the checks carried out. Whilst the business unit has access to all these files (one for each insurance risk), these documents/data types have not been used for the Data Migration phase because every file has a different structure. Therefore, it cannot be processed through a system. As a consequence, this factor requires a large amount of time whilst processing over 4000 documents, manually.

Risks level register The business unit manages an excel file which contains a registry of level of risk for every type of risk for each specific country. The insurer uses this register during the underwriting phase.

The steps taken during the "Collecting, Cleaning and defining data" process are as follows:

1. Collecting all documents and data previously presented
2. Loading all Aviation Logs Files on a temporary table called `t_import_aviation_log`;
3. Loading the Claim Reports Extraction on a temporary table called `t_import_claim_report`;
4. Loading the Policy Reports Extraction on a temporary table called `t_import_policy_report`;
5. Cleaning and defining each and every column by execution of different activities. All cleaning and defining data activities type are defined in section 4.2.1;

4.2.1 Cleaning and defining data activities

The most useful Oracle functions for these activities are:

1. `SUBSTR( column, start_position [, length ] )`
   The Oracle/PLSQL SUBSTR function allows us to extract a substring from a string.

2. `TRIM(string1)`
   The Oracle/PLSQL TRIM function removes all specified characters either from the beginning or the end of a string.

3. `REPLACE( string1, string_to_replace [, replacement_string] )`
   The Oracle/PLSQL REPLACE function replaces a sequence of characters in a string with another set of characters.
   This function is very useful as to standardize name and acronyms. For example the follow strings "LLC", "L L C" , "L.L.C." have the same information.
   This function can be used to replace the string 'L.L.C.' with a string 'LLC' with the command `REPLACE(column,'L.L.C.','LLC')` .

4. `UTL_MATCH.edit_distance_similarity(columnA, columnB)`
   This function calculates the number of insertions, deletions or substations required to transform a string-1 into a string-2, and returns the Normalized value of the Edit Distance between two Strings. The value is typically between 0 (no match) and 100 (perfect match).
   This function is very useful to intercept a list of strings that refer to the same information. For example the follow strings "BAHAMAS AIR AND BAHAMAS AIR HOLDINGS LIMITED - APP LINESLIP", "BAHAMAS AIR HOLDINGS LIMITED", "BAHAMAS AIR LTD" are different, but refer to the same aviation company "BAHAMAS AIR".
   This function can be used with others oracle functions like the following command, where it has been removed the recurrent sub-string 'AIRLINES' before calculating the similarity distance.

```
UTL_MATCH.edit_distance_similarity(
                              trim(replace(a.name,'AIRLINES',’’)),
                              trim(replace(b.name,'AIRLINES',’’))
                                   )
```

5. `TO_DATE(column,'dd-MON-YY')`
   The Oracle/PLSQL TO_DATE function converts a string to a date. Oracle[10]) whilst defining different date formats like "dd/mm/yyyy", "dd-MON-YY", "mm/dd/yyyy", "dd.mm.yyyy".
   This function is very useful to intercept date formats like "27-OCT-22", "27-10-22", "27/10/2022", "27.10.2022".

6. `TO_NUMBER( string1 [, format_mask] [, nls_language] )`
   The Oracle/PLSQL TO_NUMBER function converts a string to a number. Oracle[10] whilst defining different number formats.
   This function is very useful to intercept date format like "99200100.55", "99,200,100.55", "99.200.100,55", "992.0010055E5".

On table 4.1 all activities executed for each column of table called `t_import_aviation_log` are summarized.

Table 4.1

Name	Activities	Results
POLICY_NUMBER	- check the format of the policy code with functions n. 1, 2 - cross check with the values on t_import_policy_report - cross check with the values on t_import_claim_reportand - correct errors founded with the help of the business unit and the utilization of the extractions from BANCS.	Reduce the number of "policy number" values from 4670 to 4618. Corrected all typing errors and invalid policy numbers.
INSURED_NAME	- Intercept the values that refer to the same insured company with the functions n. 1, 2, 3, 4 and give them the same value[3] - cross check with the values on t_import_policy_report - analyze and correct errors founded both with the help of the business unit and the utilization of the extractions from BANCS.	Reduce the number of "insured name" values from 4670 to 1540 whilst correcting all typing errors and invalid values.
CLASS	- Analyse the values that refer to the same information with the functions n. 1, 2, 3, 4 - Identify and define a set of enumerated values with the help of the business unit - Identify a discrete number of enumerated values	Reduce the number of "class" values from 140 to 20 discrete values whilst correcting all the typing errors and invalid values.
SOURCE	- Analyze the values that refer to the same information with the functions n. 1, 2, 3, 4 - Identify and define a set of enumerated values with the help of the business unit	Reduce the number of "source" values from 53 to 6 discrete values whilst correcting all typing errors and invalid values.
INCEPTION	- Identify all typing errors, format errors[4] and the invalid values with the functions n. 1, 5 - cross check with the values on t_import_policy_report - Assign a value to all null records whilst retrieving the information from the values on t_import_policy_report	Correct all date strings and convert to a date variable

Continued on next page

[3]For example the following values "BAHAMAS AIR AND BAHAMAS AIR HOLDINGS LIMITED - APP LINESLIP", "BAHAMAS AIR HOLDINGS LIMITED", "BAHAMAS AIR LTD" are different but refer to the same aviation company "BAHAMAS AIR".

[4]For example the following date strings "27-OCT-22", "27- 10-22", "27/10/2022", "27.10.2022" refer to the same date

Table 4.1 – continued from previous page

Name	Activities	Results
EXPIRY	- Identify all typing errors, format errors and invalid values with the functions n. 1, 5 - cross check with the values on t_import_policy_report - Assign a value to null records whilst retrieving the information from the values on t_import_policy_report	Correct all date strings and convert to a date variable
BROKER_INSURER	- Intercept the values that refer to the same broker with the functions n. 1, 2, 3, 4 and give them the same value[5] - cross check with the values on t_import_policy_report - analyze and correct errors founded both with the help of the business unit and the utilization of the extractions from BANCS.	Reduce the number of "broker insurer" values from 344 to 128 whilst correcting all typing errors and invalid values.
RISK_DOMICILE	- Intercept the values that refer to the same country with the functions n. 1, 2, 3, 4 and give them the same value[6] - cross check with the values on t_import_policy_report - cross check with the values on the 'Risks level register' - Assign a value to all null records whilst retrieving the information from the values on t_import_policy_report	Reduce the number of "risk_domicile" (country) values from 831 to 198 discrete values.
MENA_NON_MENA	- Covert string values with two possible options 'MENA' or 'NON MENA' - Assign a value to all null records whilst retrieving the information from the column 'RISK_DOMICILE'	Reduce all values to two possible options.
USD_EUR	- Define a discrete set of values (16 different values) with the change rate - Assign a value to all null records whilst retrieving the information from the values on t_import_policy_report	Corrected all typing errors and invalid values.
HULL_M_A_VALUE	- Identify all typing errors, format errors and invalid values with the functions n. 6 - correct errors founded both with the help of the business unit and use of the extractions from BANCS.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.
CSL	- Identify all typing errors, format errors and invalid values with the functions n. 6 - correct errors founded with the helps of the business unit.	Corrected all typing errors, format errores and invalid values. Convert a string value to a number variable.

Continued on next page

[5]For example the following values "ACE INSURANCE BROKERS INDIA PVT. LIMITED", "ACE INSURANCE BROKERS LLC", "ACE INSURANCE LTD" are different, but refer to the same aviation company "ACE INSURANCE BROKERS".

[6]For example the following values "GRAND CAYMAN", "GREAT CAYMAN", "The Cayman Islands" are different but refer to the same country "Cayman Islands".

Table 4.1 – continued from previous page

Name	Activities	Results
PA	- Identify all typing errors, format errors and invalid values with the functions n. 6 - correct errors founded with the help of the business unit.	Corrected all typing errors, format errors, invalid values. Convert a string value to a number variable.
AVN_52	- Identify all the typing errors, format errors, invalid values with the functions n. 6 - correct errors founded with the help of the business unit.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.
GROSS_PREMIUM_100	- Identify all typing errors, format errors and invalid values with the functions n. 6 - cross check with the values on t_import_policy_report - correct errors founded both with the help of the business unit and the use of BANCS extractions.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.
WRITTEN_LINE	- Identify all typing errors, format errors and invalid values with the functions n. 6 - correct errors founded with the help of the business unit.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.
SIGNED_LINE	- Identify all the typing errors, format errors and the invalid values with the functions n. 6 - cross check with the values on t_import_policy_report - correct errors founded both with the help of the business unit and the use of BANCS extractions.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.
U_YEAR	- Identify all typing errors, format errors and invalid values with the functions n. 1, 5	Correct all date strings and convert a string value to a date variable
COSTS	- Identify all typing errors, format errors and invalid values with the function n. 6 - correct errors founded with the help of the business unit.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.
OFFERED_DIFF	- Identify all typing errors, format errors and invalid values with the function n. 6 - correct errors founded with the help of the business unit.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.
FINAL_DIFF	- Identify all typing errors, format errors and invalid values with the function n. 6 - correct errors founded with the help of the business unit.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.
LEADERS_FEES	- Identify all typing errors, format errors and invalid values with the function n. 6 - correct errors founded with the help of the business unit.	Corrected all typing errors, format errors and invalid values. Convert a string value to a number variable.

4.3 Data migration

The database created and presented in Figure 4.1 has been populated taking the following steps:

- Create a PL/SQL Oracle Procedure to populate a part of the tables whilst using as data source the temporary table called `t_import_aviation_log` (after the cleaning and defining data activities).
- Execution of the PL/SQL Oracle Procedure and check the results.
- Populate:
  - the currency table with the list values defined during the cleaning and defining data activities.
  - the countries table with the list values defined during the cleaning and defining data activities.
  - the claims table by using as data source the table called `t_import_claim_report`
  - the risk levels table by using as data source the 'Risks level register'.
- Enable all the oracle constraints and checks related to the populated tables.
- Generate a set of data extractions ('List of the Broker Insurer', 'List of the Insured Companies' etc..) that has been checked and validated by the business unit.

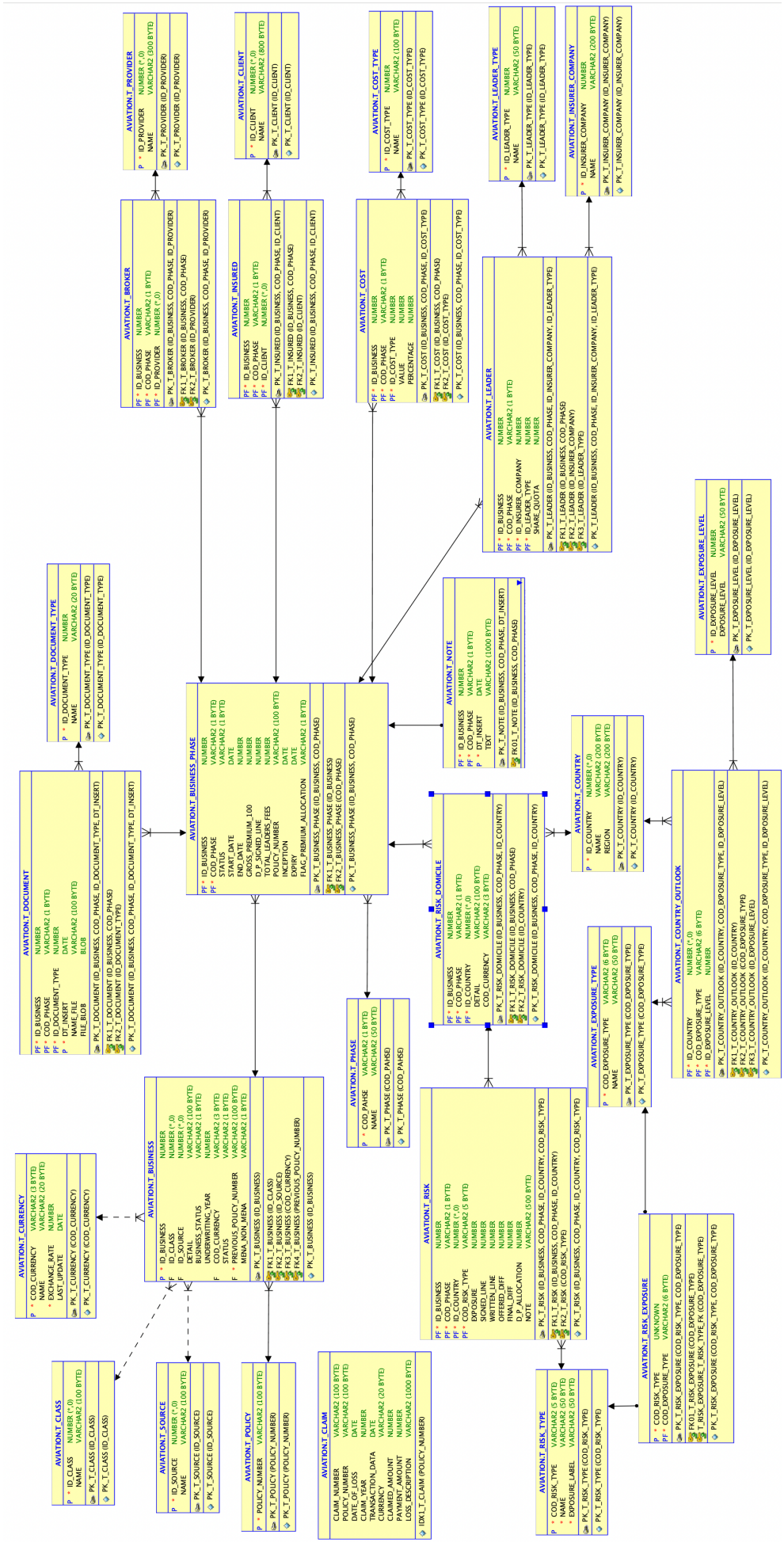

Figure 4.1: Oracle Database schema generated by the SQL Developer application

Chapter 5

The business problems

As mentioned on Chapter 1 section 1.2 there are two aspects which influence issues and demands management tasks: Timing and Past knowledge. All problems that should be resolved are described in this chapter starting from these aspects, of course. In addition, it is defined the dataset which can be used both by AI and probabilistics models.

5.1 Business problem A: subset of pre-existing cases

Due to the severe work-load managed in a ***limited period of time*** and the fact that the information data structure does not permit a fast and efficient comparison, the Underwriter is not in a position to compare past analyses to the new risk offered. In other words the Underwriter has to replicate the whole evaluating process for each and every risk constantly. It seems that the underwriting process has ***no past knowledge***.

This is the main reason why the business unit is looking for a tool capable to obtain a subset of pre-existing comparable cases to the case under underwriting review/analysis. This tool should be able to help underwriters as to apply past considerations to a new risk under evaluation.

The first business problem under evaluation on this thesis work is to find a way for selecting a set of past insurance risks regarding to a specific insurance risk under analysis.

This tool should help the insurer during the underwriting phase.

The business unit is unable to identify a set of filters that allow an extraction of a past insurance risks set similar to a specific insurance risk under analysis. For this reason a simple sql query on database does not suffice. This may require the definition of a specific sql query for each specific cluster type of insurance risks, at least.

At the same time, the business unit is not able to classify each and every past

insurance risk whilst dividing those items by cluster as a lot of data contributes to such an outcome as for instance: Class of business, insured, type of risk, level of risk for a specific type of risk on a specific risk domicile , etc...

Figure 5.1: Useful dataset regards to the Business Problem A

5.1.1 Useful dataset

A data set of interests has been identified with the assistance of the business unit. This set of data may be used to select a subset of similar insurance risks.

A data set of interests is listed as below:

- **All fields highlighted in orange in the mock-up on Figure 5.1.** These data coincide with the data entered by the user during the pre-underwriting phase. These fields are the only available data to the insurer at beginning of the underwriting phase; hence and they represent the only available data which can be used to select a subset of similar insurance risks.
- **All levels of risk related to the policy.** A risk level is extracted from a set of tables which map a level of risk both to the Type of risk and the Risk domicile.
  For instance, looking the case represented in Figure 5.1 this policy has:
  - the level of risk related to the risk "personal accident" equals to zero as this policy does not has a "personal accident" risk
  - the level of risk related to the risk "combined single limit" equals to "Medium risk" as the level of risk extracted from the tables for a "combined single limit" risk domiciliated in Tunisia is "Medium"

5.2 Business problem B: Data entry errors

Due to the severe work-load concentrated in a ***limited period of time*** the probability of data entry errors become very high. This is the reason behind the business unit requests to obtain a tool which may help the underwriter to avoid/ prevent data entry errors.

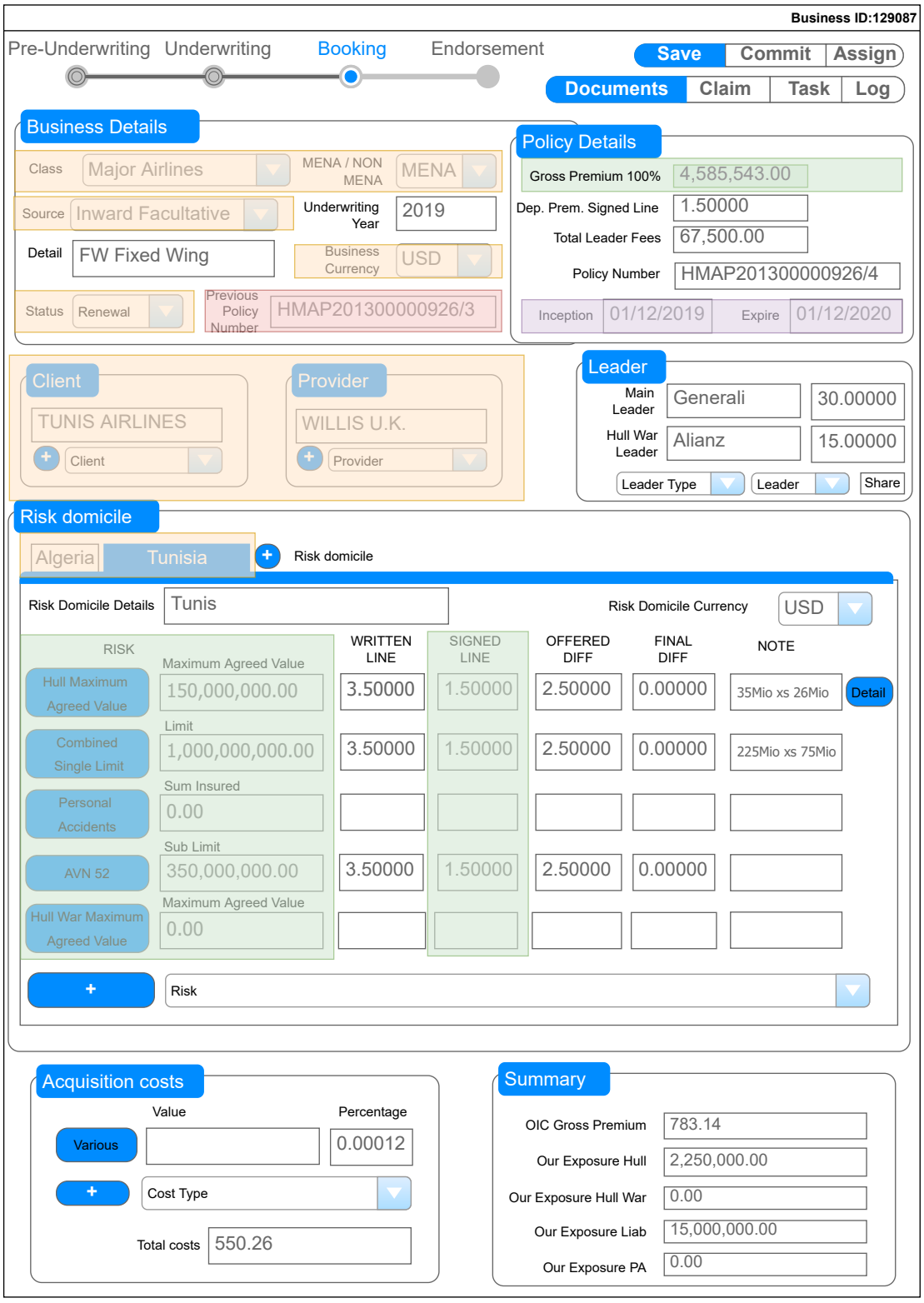

Figure 5.2: Useful dataset regard to the Business Problem B

The second business problem under evaluation on this thesis work is to find a way (a system) for helping the user to prevent data entry errors.
With the assistance of the business unit the most important fields where the highest number of data entry errors happened historically, have been identified. Those fields are highlighted in green on Figure 5.2. All these fields should be checked by a system in order to prevent data entry errors.

5.2.1 Useful dataset

With the assistance of the business unit it has been picked up the following data which may be useful to this task :

- All fields highlighted in orange on Figure 5.2 are the data already available in the system; whilst all fields highlighted in green are filled in by the underwriters. The orange fields should have a lower data entry error probability since the user will select the value from a predefined list (combo-box) in the new system.
- The field highlighted in red on Figure 5.2 is very important as it can help the system to consider the past policies related both to the same client and the same insurance risk. In many cases, the values of the green fields do not change from the old insured policy to the new one.
- The two fields highlighted in purple on Figure 5.2 are two dates which should be checked as a combination of both. In other words the system shoul be able to check the number of days between these two dates, not the specific day identified by a date.

Chapter 6

AI and Probabilistic models

In this Chapter both the AI and probabilistic models used to solve the business problems presented in the previous chaperts are analyzed and described in detail.

6.1 Solve The Business Problem A

A possible way to resolve the "Business Problem A" described on Chapter 5.1 is to **make a recommender system**.

6.1.1 Approach

As stated in the previous chapter, a simple approch to realize this system by using a set of SQL Queries has not been considered. Whilst, there are various features to take into account; the Businnes Unit is not able to provide a proper set of rules as to segment the insurance business. For instance, the values of the feature "Gross Premium 100" cannot be subdivided in a set of bands (As "low level Gross Premium 100" , "medium level Gross Premium 100" , etc...) since a set of bands can be properly defined for a set of insurance business, but can be incorrect for another set at the same time.

The schema on Figure 6.1 represents different types of machine learning algorithms and a Recommender System is a perfect use case example as to apply a machine learning algorithm.

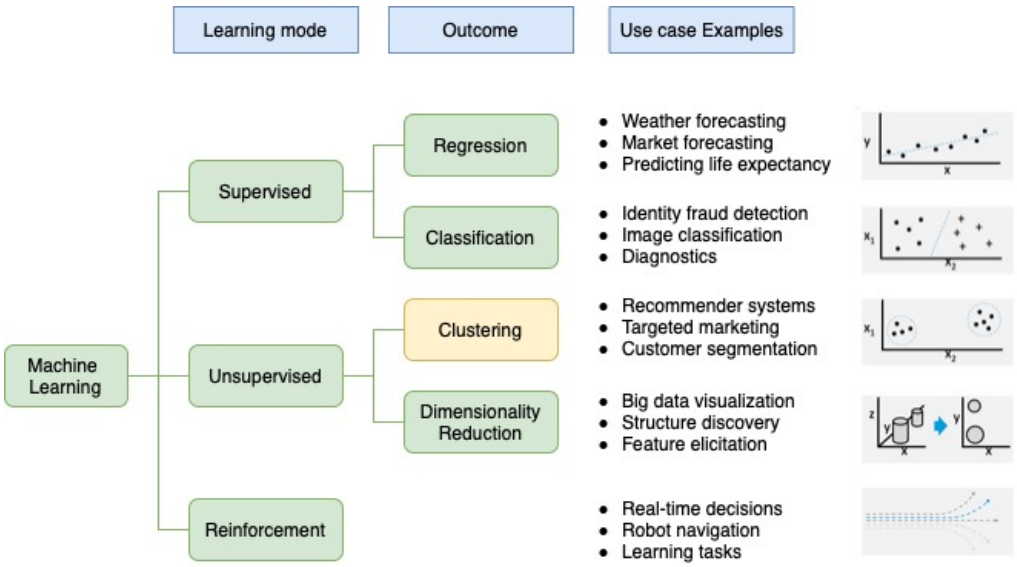

Figure 6.1: Types of machine learning algorithms

6.1.2 The AI Model/Algorithm

The Recommender System has been realized by using a **clustering unsupervised machine learning algorithm** as recommended by the schema on Figure 6.1 .
In our case, an Inductive machine learning is needed. This equals to build a model with some data which can then be applied to new instances.
The data used by the clustering algorithm has already been described in the chapter 5.1.1 and shown in figure 5.1.

On this thesis work the operational approach, used to identify the best clustering algorithm as to realize a good AI model which allows to solve the business problem, is focused on the analysis of different clustering unsupervised algorithms.
The python library called scikit-learn[11] let compare easily different types of algorithms[12] that are reported on Table 6.1.

A specific emphasis has been given to the **K-Means algorithm** as it is recongnized as the most known and used clustering method.
The literature proposes various extensions of k-means. Whilst the algorithm k-means is an unsupervised algorithm of clustering in pattern recognition and machine learning; this is still influenced by the initialization with a number of clusters. Hence, this algorithm cannot be exactly defined as an unsupervised clustering method. In literature there are some articles [13] that propose a novel unsupervised k-means (U- k-means) clustering algorithm which automatically finds an optimal number of clusters without giving any initialization and parameter selection.
In our case, this is all positive as the business unit is not able to identify the clusters, but they have an expectation to separate all the insurance business in a set of

200/300 groups. This evaluation is made on the base of the technical knowledge of the Underwriters.

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large n_samples, medium n_clusters with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters, inductive	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry, inductive	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry, inductive	Distances between points
Spectral clustering	number of clusters	Medium n_samples, small n_clusters	Few clusters, even cluster size, non-flat geometry, transductive	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters or distance threshold	Large n_samples and n_clusters	Many clusters, possibly connectivity constraints, transductive	Distances between points
Agglomerative clustering	number of clusters or distance threshold, linkage type, distance	Large n_samples and n_clusters	Many clusters, possibly connectivity constraints, non Euclidean distances, transductive	Any pairwise distance
DBSCAN	neighborhood size	Very large n_samples, medium n_clusters	Non-flat geometry, uneven cluster sizes, outlier removal, transductive	Distances between nearest points
OPTICS	minimum cluster membership	Very large n_samples, large n_clusters	Non-flat geometry, uneven cluster sizes, variable cluster density, outlier removal, transductive	Distances between points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation, inductive	Mahalanobis distances to centers
BIRCH	branching factor, threshold, optional global clusterer.	Large n_clusters and n_samples	Large dataset, outlier removal, data reduction, inductive	Euclidean distance between points
Bisecting K-Means	number of clusters	Very large n_samples, medium n_clusters	General-purpose, even cluster size, flat geometry, no empty clusters, inductive, hierarchical	Distances between points

Table 6.1: Clustering algorithm on scikit-learn library

6.2 Solve The Business Problem B

There are various methods as to resolve the "Business Problem B" described on Chapter 5.2, the initial steps are:

1. assign a definition of error data;
2. explore different approaches as to prevent data entry errors and/or analyze the errors in the currentd context;

6.2.1 Error data

Error data refers to the data which, to the contrary of the specific rules and laws in the application, shows characteristics of manifest error whilst it is determined. This means that the data themselves do not meet the application needs, or the data association does not meet the normal logic.
The mathematical description of error data can be expressed in the form below: Supposed $S = (U, R, V, f)$ is a knowledge expression system, where U is the domain, R is the attributes set, V is the attribute values set, $f : UXR \rightarrow V$ is a information function, which specifies the attribute value of each element in the U. The mathematical description of the error data should be defined as follows:

1. Set a_k is the attribute value of attribute A of the k^{th} record. If there is $a_k \notin V_A$ the data values of a_k is incorrect.
2. Set a_k is the attribute value of attribute A of the k^{th} record, b_k is the attribute value of attribute B of the k^{th} record. If the attribute B acquires value from b_k , the attribute A cannot be a_k , the logic error exists between the data of a_k and b_k

6.2.2 Approaches to prevent data entry errors

There are various approaches to prevent data entry errors such as:

1. **Error data detection algorithm based on rules**
   This means:
   - Define a set of rules to identify data errors such as: missing values, format errors, etc...; [14][15]
   - Introduce check digits on data values and error detecting codes as to protect fixed length decimal data as an example. [16]
2. **Error data detection algorithm based on statistics** Realize a system capable to detect erraneous input values that look statistically abnormal such as: outlier detection, etc...

3. **Improving data quality with dynamic forms** [17] Realize a system with a probabilistic model over the questions of the form. The system applies this model at each and every step of the data-entry process as to improve data quality. It adapts dinamically the form to the values being entered by providing real-time interface feedback during the data entry, re-asking questions with doubtful responses, and simplify questions by re-formulating the same.

4. **Error detecting and tagging system** [18] [19] Fist of all classifying data entry errors types, define a specific startegy and realyzing a system capable to detect erroneous input values. The system should be capable to detect erraneous input values that look normal statistically, but abnormal in the specific context.

The first approach is the "standard/basic" approach to prevent data entry error used on web applications. The data knowledge required is mainly limited to type of data that the form/application manage. It allows to incercept absolute error on data mainly, for example: the "The Gross Premium 100" value cannot be a char string, the "The Gross Premium 100" amount cannot be a negative number.

The second approach uses statistic methodologies to intercept errors on values looking the past data managed. The data knowledge requires enough past data as to assure good statistics. It gives the chance to intercept errors on values for a specific data as, for instance, the "Gross Premium 100" amount cannot be equal around 100 dollars since "there are no "Gross Premium 100" amounts which equal to 100 dollars in the past data".

The business unit has classified the majority of the usual data entry errors like relative errors.
This means that the value of a specific field could be both "admitted" whilst looking at the value in absolute way; and "wrong" in a relative way considering the context and the insurance business's past history for a certain client.
As an example, if a value of the field "Gross Premium 100" equals to USD 70 Millions is admitted because USD 70 Millions is acceptable for a "Gross Premium 100"; an amount of USD 100 is certainly wrong. At the same time, this amount of USD 70 Millions can be considered correct if the risk refers to CAAC (China Airlines group) or just wrong if the risk refers to Ryanair which has policies with Gross Premium of around USD 6 Millions.

Due to this, both the first and second approach are not good enough to reach the desired target.

The third approach is very innovative, but requires a large data knowledge managed by the application. By recalling the previous example, the system after receiving the data related to an amount of "gross Premium 100" which equals to USD 70 Millions should be able to:

- check the correctness of the amount by utilizing other data previously received from the user
- request other details correlated to the specific insurance business to the underwriters, such as the number of aircraft covered under the risk, once/ if the data previously received were not sufficient to complete the first step.

Whilst this approach should be able to prevent the vast majority of error types cited by the business unit, its implementation is not simple at all. In addition, a specific control system should be realized for each and every field because it is strictly related to the nature, type and context of the data.

The fourth approach (to the student's opinion) is the most linear and innovative approach. The first step implies the identification of all possible data entry error types whilst adopting a specific strategy for each of them as to prevent and/ or tag the error. A system implemented with this approach should be able to detect erroneous input values that looks like normal statistically; but abnormal in the current context. (for example a system to detect errors based on Bayesian network to learn dependency relationships among fields [18]).

As a consequence, the fourth approach has been chosen to resolve the business problem B.

6.2.3 Initial idea

During the bibliographic research phase of this thesis work the student has found an article [19] which tries to resolve the data entry errors problem with the approach number 4 applied on medicals data. The student found on this article a similarity between the medical data set and the insurance business data set on this specific field (space and aviation risks). The similarity here-with highlighted has been represented on figure 6.2 then discussed with the business unit. The similarity can be summarized as follows:

- to a group of insurance business corresponds a heterogeneous set of people
- to a homogeneous group of people (same age, same sex, similar medical history, etc.) corresponds a group of similar insurance business (similar type of insurance risk). As stated by the business unit, Wizzair, Ryanair and Easyjet are similar

risks; whilst Qatar Airways rather than Singapore Airlines represent a different type of risk.

- to a set of medical records related to the same person during the course of one's life corresponds an insurance policy and its annual renewals.
  As stated by the business unit each and every policy and its renewals are similar: every renewal may differ by a small variation from the previous renewal. These variations can be considered as a natural evolution of the policy during the course of the years.

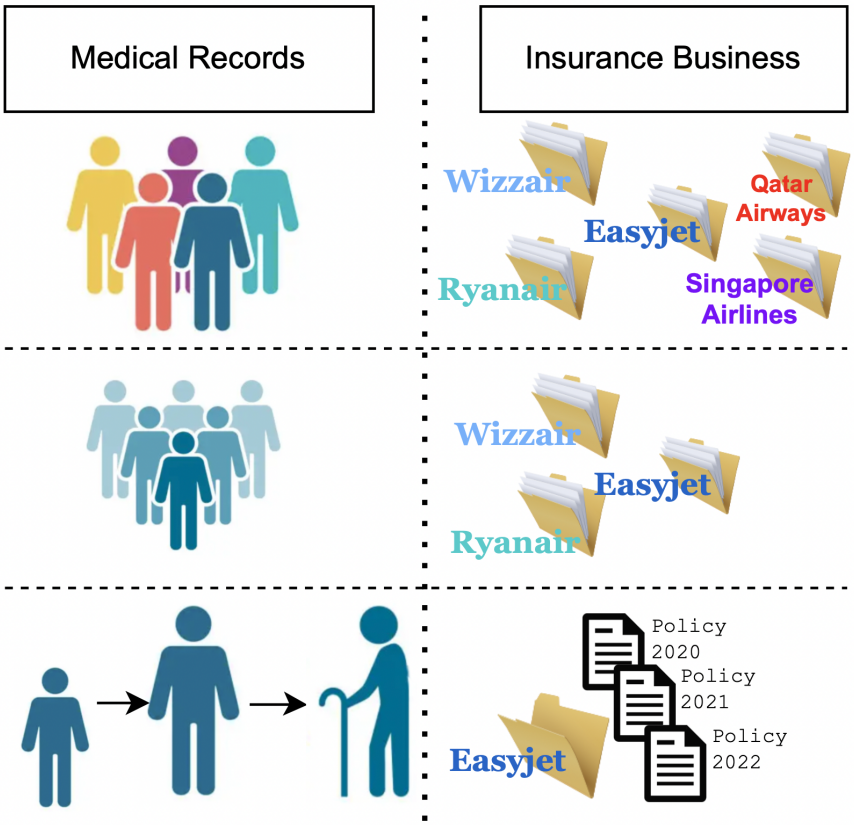

Figure 6.2: Similarity between the medical data set and the insurance business data set

The business unit agreed on this intuition. Furthermore, the underwriters declared that around 85% of their annual insurance business in the Aviation and Space arena consists of policies renewal.
Taking into account the following bullet points:

- the similarity found and confirmed by the business unit;
- the fact that 85% of the annual insurance business consists of renewals;
- the problem to identify group of similar insurance business has been affronted with problem A, already;
- the model proposed in the cited article [19] has better performance in comparison

to other statistical model such as Hampel X84 and Bayesian Network;

- the model proposed in the cited article [19] generates better results than other methods which assume that the input value for a field is based solely on his historical records;
- by using tagging mechanism, such as the one proposed in the cited article [19], has been shown the decrease of certain types of errors[20];

the initial idea to resolve the problem B starts from the novel proposed in the cited article[19].

6.2.4 Approach to business problem solving

A way to resolve the "Business Problem B" described on Chapter 5.2 is summarized as follows:

1. Identify/Analyze Data entry error types
2. Define a strategy to prevent or tag each and every type of error
3. Make a data entry errors detecting system for the type of error that cannot be prevented

6.2.5 Data entry error types

Identify/Analyze different types of data entry error

On Figure 6.2 are reported the 7 data entry error types identified.

Level 1: Incorrect Format and Missing	L1-1: Incorrect Format Data
	L1-2: Missing Data
Level 2: Out of Range	L2-1: Out of normal range
	L2-2: Out of population trends
	L2-3: Out of personal trend
Level 3: Inconsistent	L3-1: Personal Inconsistent
	L3-2: Population Inconsistent

Table 6.2: Levels of Data Entry Errors

L1-1 errors can be easily controlled with prior knowledge of data formats.
Constraints can be set for each and every field.
As an example, numbers can be entered into "Gross Premium 100" field, only.

L1-2 errors can be easily controlled with prior knowledge about data format.
Constraints can be set for each and every field.

For instance, the "Limit" field has to be mandatory if the insurance business has a "Combined Single Limit" risk, only.

L2-1 errors can be easily controlled as long as there is a knowledge of the normal range of each and every field.
As an example, the range for the "Signed Line" field goes from 0 to 100.

L2-2 errors can be detected by using statistics for a single attribute [1].
For example, the amount of "Gross Premium 100" field cannot be 100 dollar because there are no policies with this premium amount.

L2-3 errors can be detected by using statistics for a single attribute limited to the data of the past renewals related to the same risk.
As an example, the amount of "Gross Premium 100" field cannot have an increment of 5% in comparison to the previous year value if this type of risk remains unchanged for years. On the other hand, this can be possible for a risk related to an aviation Client that is growing up year by year and their trend shows an increase of the "Gross Premium 100" value year by year.

L3-1 errors refer to inconsistent cases for a specific insurance business.
For example, a renewal policy can have a "Gross Premium 100" amount reduction if some of the limits specified for the type of risks have a reduction but, it will not be acceptable if all the risk limits remain the same.

L3-2 errors refer to inconsistent cases in comparison to similar insurance business.
For instance, a risk/policy domiciled in a certain country which does not have "Hull War maximum agreed value" risk might be possible, but it looks inconsistent if all other policies domiciled in the same country have this interest covered (this could mean that this country has an ongoing conflict).

Define a strategy for each data entry error types

L1-1 , **L1-2**, **L2-1** errors can be prevent by putting constraints on the application form.

L2-3 , **L3-1**, **L3-2** errors can be detected with a system capable to detect errors on values that look like statistically normal, but are abnormal in the current context, also.

An hybrid model that uses both AI and a probabilistic/statistics approach to prevent data entry error on an information system is presented on the next section.

[1] For instance, this type of error can be detected by using univariate outlier detection technique called Hampel X84 [21].

6.2.6 Data entry errors detecting system

In this section, it is introduced an error detecting and tagging machanism in real time as to address these challenges based on historical data and dependency relationships. The entire framework consists of three main components:

1. **Find a similar insurance business** The error detection is based on historical datasets. For a single insurance business, the historical dataset should be composed of insurance business which is equivalent to the case evaluation. As it is difficult to find a group with the same characteristics, the system will detect errors on the base of a dataset arising from insurance business which shows similar conditions.

2. **Learn a probabilistic model** The probabilistic model will be used to calculate the input value probability for each and every field that has to be checked in order to prevent data entry errors.
   The model is based both on the dataset of similar insurance business and the current insurance business.
   This model is designed on two assumptions:

   - the current insurance business' input value for a field depends upon its historical data (past renewals for the same risk)
   - the input valued for a field depends upon the other fields values.

   The system will combine the influences arising from these two aspects.

3. **Return a Tag** The error tagging is based on the results originated from the probabilistic model. If an input value probability for a field is relatively high, then the value has a higher chance to be a correct figure; therefore a "Normal" tag will be returned. Otherwise, if the probability is relatively low, then the tag will be "Suspicious". Hence, the system will use a threshold in order to decide which tag should be returned ("Normal" or "Suspicious").

The data entry errors tagging system is made by five stages:

1. selecting a classification model;
2. find a similar insurance business group;
3. building a probabilistic model;
4. setting a threshold;
5. returning tags;

The detecting and tagging framework is high-lighted in Figure 6.3. A data entry interface contains multiple fields. After an user inputs a value for a field, a tag

("Suspicious" or "Normal") will be returned. This system classifies the input values of each and every field as "Suspicious" or "Normal"; then provides this information as a feedback to the user. If the "Suspicious" tag occurs, then this reminds to the user to check the input value. As a consequence the data entry error will be reduced.

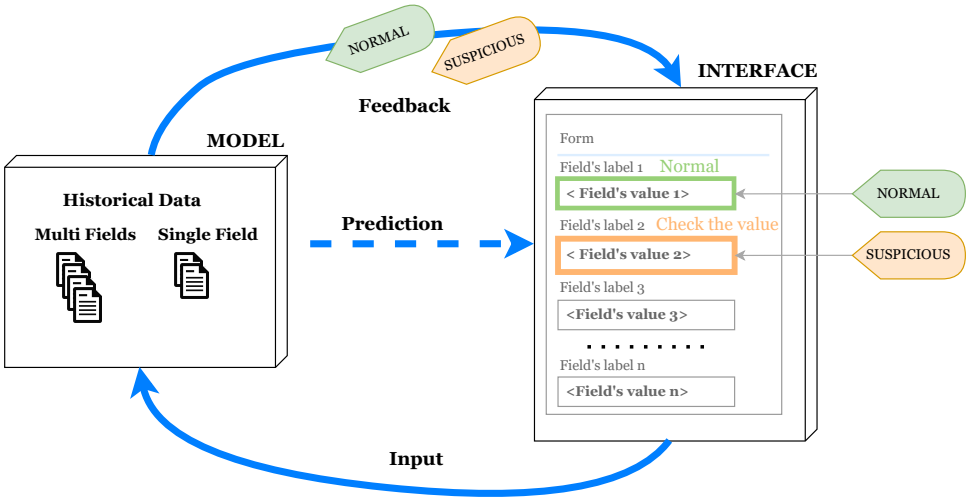

Figure 6.3: Error Detecting and Tagging Framework

6.2.7 Error Detecting Process overview

The error detecting process, based on error detecting framework as presented previously, is shown on Figure 6.4.
The data used by the error detecting process has already been described in the chapter 5.2.1 as shown in the figure 5.2.
A form is used to collect a set of specific types of information which are divided into two categories:

- **"indipendet fields" or "basic fields"** which are considered as fixed during the error detection process;
- **"dependent fields" or "fields to be checked"** which will be collected and flagged as "suspicious" or "normal";

The "basic fields" correspond to the fields high-lighted in orange in figure 5.2 whilst the "fields to be checked" correspond to the fields high-lighted in green in the same figure.

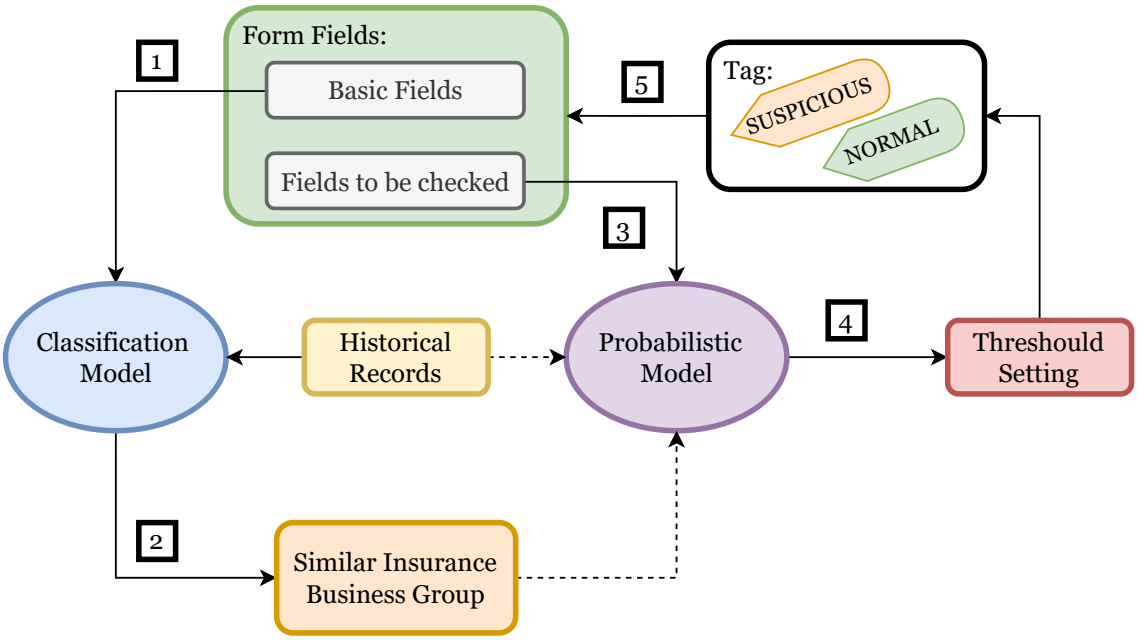

Figure 6.4: Error Detecting Process

As shown in figure 6.4, the error detecting process consists of five steps:

1. Insurance business records are categorized into groups. The categorization is based on information contained on basic indipendent fields (fields highlighted in orange in figure 5.2) related both to the current insurance business and other historical records. This is realized by using a classification model.
2. The classification model produces a group of similar insurance business.
3. The probabilistic model is designed both around "Fields to be checked" and historical records from a group of similar insurance business.
4. A threshold is defined for the probability of input value.
5. The system returns a "normal" or "suspicious" tag on the base of the results arising both from the probabilistic model and the threshold setting.

6.2.8 Error Detecting Process: Classification Model

The classification model is realized following the same approach used to resolve the business problem A which has been discussed in the chapter 6.1.
The data types used by the clustering algorith corresponds to the fields high-lighted in orange in the figure 5.2.

6.2.9 Error Detecting Process: Probabilistic Model

Data entry error detecting method

Gived a data entry interface I, let $F = \{F_v, ..., F_j, ..., F_k\}$ be a set of input fields on the interface I. For example, the fields used to enter the "Gross Premium 100" value can be represented as $F_{GrossPremium100}$.
For a single field F_v it is used a sequence of values $v = \{v_1, ..., v_{i-1}, v_i\}$ to represent all historical records related to the same risk (past renewals), where v_i is the current valued entered by a user for the filed F_v, $\{v_1, ..., v_{i-1}\}$ is the historical record sequence for the field F_v and v_{i-1} is the valued entered for the field F_v at the $(i-1)^{th}$ time.

***Assumption**: the current insurance business input value for a field depends on historical input values recorded for this field in past renewals. Abnormal changes of input values in comparing with the historical records shall get appropriate attention.*

In probability theory, such as sequentially dependent process can be described by a Markov model. According to 1^{th} order Markov assumption, the current value for field F_v at i^{th} time depends only on the value for the same field at $(i-1)^{th}$ time. Therefore, for a giving input value v_{i-1} to predict the probability of the value v_i for the field F_v at the current time. The probability can be represented as $q_m(v_i|v_{i-1})$. Similarly, we can have a probability according to 2^{th} order Markov assumption as $q_m(v_i|v_{i-1}, v_{i-2})$.
But for a new insurance business (non a renewal insurance business) there are no historical records for the value of field F_v. So we can assume that the current value v_i for the field F_v at i^{th} time does not depend on previous historical data. Therefore the predicted probability can be represented as $q_m(v_i)$.

Our goal is to predict the probability of an input value v_i for the field F_v, which can be represented as $q_M(v_i)$.
Using 1^{th} order Markov assumption, 2^{th} order Markov assumption and a prior probability, we can calculate $q_M(v_i)$ as in Equation 6.1:

$$\begin{aligned} q_M(v_i|all\_conditions) = \\ \alpha_3 \times q_m(v_i|v_{i-1}, v_{i-2}, all\_conditions) \\ +\alpha_2 \times q_m(v_i|v_{i-1}, all\_conditions) \\ +\alpha_1 \times q_m(v_i|all\_conditions) \end{aligned} \tag{6.1}$$

all_conditions means that the probability is calculated on the base of a dataset of similar insurance business, only. In the following equations it will be obmited *all_conditions* in each term assumimng that they have been taken into consideration.

The three parameter probabilities α_1, α_2 and α_3 are used to balance the three probability and $\alpha_1 + \alpha_2 + \alpha_3 = 1$.
Given a set of records, it is possible to estimate each part of the probability according to Equation 6.2, Equation 6.3 and Equation 6.4, respectively.

$$q_m(v_i|v_{i-1}, v_{i-2}) = \frac{N(v_{i-2}, v_{i-1}, v_i)}{N(v_{i-2}, v_{i-1})} \tag{6.2}$$

$$q_m(v_i|v_{i-1}) = \frac{N(v_{i-1}, v_i)}{N(v_{i-1})} \tag{6.3}$$

$$q_m(v_i) = \frac{N(v_i)}{N(all)} \tag{6.4}$$

For Equation 6.2 and Equation 6.3 is the occurance number of input value v_{i-2} for field F_v at the $(i-2)^{th}$ followed by the input value v_{i-1} for field F_v at the $(i-1)^{th}$ time for a single insurance business (current and past renewals).
Similarly, $N(v_{i-2}, v_{i-1}, v_i)$ is the occurrence number of input value in a sequence of $\{v_{i-2}, v_{i-1}, v_i\}$ for the $(i-2)^{th}$, $(i-1)^{th}$ and i^{th} time for a insurance business

For Equation 6.4, $N(all)$ is the total number of values for field F_v for historical records from all similar insurance business. Similar insurance business means a group of insurance business which are similar to the current. $N_(v)$ is the occurence number of v_i for field F_v from all historical records based on all similar insurance business.

The Equation 6.1 only uses the historical records from a *single* field. However, there are dependency relationships among different fields on a data entry interface I. The relationships among different fields can be obtained based on prior knowledge from experts (the business unit in this case) or learned from historical data.
For example: the value v_i of F_v depends on the value F_w and F_u, so it is define the probability $q_B(v_i, |w_i, u_i)$. Adding the probability of $q_B(v_i, |w_i, u_i)$ to the Equation 6.1 is obtain the following Equation 6.5:

$$q(v_i) = \beta \times q_M(v_i) + \gamma \times q_B(v_i|w_i, u_i)) \tag{6.5}$$

where the two parameter β and γ corrispond to $\beta + \gamma = 1$. The value of $q_M(v_i)$ comes from the Equation 6.1. The value of $q_B(v_i|w_i, u_i)$ comes froom CPT of the learned Bayesian Network. By plugging in the above values, it is compute a final probability $q(v_i|all\_conditions)$.

Error Tagging Method

It is assumed that for each field F_v the set of possible values is finite and discrete. If the values for a field are continuous, appropriate methods for discretizing them shall be applied.
Let $V = \{v_i', v_i'', ...\}$ be the possible values for the field F_v. The current input value v_i at the i^{th} time is a member of the set. For each possible input values in the set V, it is possible to calculate the probability base on the Equation 6.5 and obtain the tag value from Equation 6.6.

$$tag(v_i) = \begin{cases} Normal & q(v_i) \geq \theta \\ Suspicious & q(v_i) < \theta \end{cases} \tag{6.6}$$

Chapter 7

Make the models

In this chapter it is described both the modelling process and the way to utilize these models. It introduces all software's frameworks needed to perform such an activity. It also describes all codes produced and analyzes all charts useful to evaluate the models.

7.1 Software's Framework

Before starting this phase ot the project, it is mandatory to prepare the correlated environment. The required software is high-lighted here-below:

- Oracle database (see on chapter 4.1)
- Python 3.9.1
  - library: sys - In order to read system parameters.
  - library: os - In order to interact with the operating system.
  - library: numpy - In order to work with multi-dimensional arrays and matrices.
  - library: pandas - In order to manipulate and analyze data.
  - library: matplotlib - In order to create 2D charts.
  - library: yellowbrick - In order to visualize elbow and silhouette plots.
  - library: sklearn - In order to perform data mining and analysis.
  - library: time - In order to calculate training time.
  - library: seaborn In order to perform data visualization
- Anaconda Navigator 2.2.0
- Tensorflow 2.5
- Jupyter 6.4.12

7.2 Initial Dataset description

The following Table 7.1 describes the initial dataset which contains data extracted from the database, directly.

Oracle objects created/utilized during this phase:

- creates the PL/SQL view named V_EXPORT_DATA to select all the data from our database.
- creates the PL/SQL materialized view named M_V_EXPORT_DATA to select all data from V_EXPORT_DATA as to perform following queries on this view, quickly.
- creates the PL/SQL view name V_CSV_DATA to convert all the data from the view V_EXPORT_DATA on string values (manage date and number conversion/format).

Table 7.1: Initial Dataset extracted from database

Feature	Feature Description	Type	Value
ID_BUSINESS	Identify in a unique way a Business	Number	ID
COD_PHASE	Identify the business phase	String	P : Pre-Underwriting - U : Underwriting - B : Booking - E : Endorsement
BUSINESS_STATUS	Business status	String	C : Close - O : Open
ID_CLASS	Business Class type	Number	ID
CLASS		String	Decode of ID_CLASS from table T_CLASS
ID_SOURCE	Business Source type	Number	ID
SOURCE		String	Decode of ID_SOURCE from table T_SOURCE
STATUS	Business status	String	N : New - R : Renewal
PREVIOUS_POLICY_NUMBER	PREVIOUS POLICY (In case STATUS = R)	String	Previous policy number (valorized only if STATUS + 'R')
UNDERWRITING_YEAR	Underwriting Year	Number	Year
MENA_NON_MENA	Mena/Non Mena	String	N : NON MENA - M : MENA
BUSINESS_COD_CURRENCY	Business Currency type (see decode on table T_CURRENCY)	String	
BUSINESS_EXCHANGE_RATE	Business Currency Exchange Rate (see decode on table T_CURRENCY)	Number	
INCEPTION		Date	Data of Inception (format dd/mm/yyyy)
EXPIRY		Date	Data of Expiry (format dd/mm/yyyy)

Continued on next page

Table 7.1 – continued from previous page

Feature	Feature Description	Type	Value
GROSS_PREMIUM_100	Gross Premium 100%	Number	Float number. (The value is expressed in the currency of the business: BUSINESS_COD_CURRENCY)
POLICY_NUMBER	Policy number	String	
FLAG_PREMIUM_ALLOCATION	FLAG Premium Allocation presence	String	T : Presence of Premium Allocation - F : Absence of Premium Allocation
D_P_SIGNED_LINE	Deposit Premium Signed Line	Number	(valorized in case of FLAG_PREMIUM_ _ALLOCATION = 'F')
ID_PROVIDER	Business Provider (see decode on table T_PROVIDER)	Number	ID
PROVIDER		String	Decode of ID_PROVIDER from table T_PROVIDER
ID_CLIENT	Business Client (see decode on table T_CLIENT)	Number	ID
CLIENT		String	Decode of ID_CLIENT from table T_CLIENT
ID_COUNTRY	Business Country (see decode on table T_COUNTRY)	Number	ID
COUNTRY		String	Decode of ID_COUNTRY from table T_COUNTRY
COUNTRY_REGION	Region of Country	String	Country Region from decode table T_COUNTRY
EXPOSURE_HMAV	Number that represents the Maximum Agreed Value of the "Hull Maximum Agreed Value" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
SL_HMAV	Signed Line of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1. (only positive value)
WL_HMAV	Writte Line of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1. (only positive value)
OF_HMAV	Offered Differential of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
FD_HMAV	Final Differential of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
DPA_HMAV	Deposit Premium Allocation of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
EXPOSURE_CSL	Number that represents the Limit of "Combined Single Limit" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)

Continued on next page

Table 7.1 – continued from previous page

Feature	Feature Description	Type	Value
SL_CSL	Signed Line of "Combined Single Limit" Risk (valorized in case FLAG_PREMIUM_ _ALLOCATION = 'T')	Number	Float number from 0 to 1. (only positive value)
WL_CSL	Writte Line of "Combined Single Limit" Risk	Number	Float number from 0 to 1. (only positive value)
OF_CSL	Offered Differential of "Combined Single Limit" Risk	Number	Float number from 0 to 1.
FD_CSL	Final Differential of "Combined Single Limit" Risk	Number	Float number from 0 to 1.
DPA_CSL	Deposit Premium Allocation of "Combined Single Limit" Risk	Number	Float number from 0 to 1.
EXPOSURE_PA	Number that represents the Sum Insured of "Personal Accidents" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
SL_PA	Signed Line of "Personal Accidents" Risk (valorized in case FLAG_PREMIUM_ _ALLOCATION = 'T')	Number	Float number from 0 to 1. (only positive value)
WL_PA	Writte Line of "personal Accidents" Risk	Number	Float number from 0 to 1. (only positive value)
OF_PA	Offered Differential of "personal Accidents" Risk	Number	Float number from 0 to 1.
FD_PA	Final Differential of "personal Accidents" Risk	Number	Float number from 0 to 1.
DPA_PA	Deposit Premium Allocation of "Personal Accidents" Risk	Number	Float number from 0 to 1.
EXPOSURE_AVN52	Number that represents the Sub Limit of "AVN 52" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
SL_AVN52	Signed Line of "AVN 52" Risk (valorized in case FLAG_PREMIUM_ _ALLOCATION = 'T')	Number	Float number from 0 to 1. (only positive value)
WL_AVN52	Writte Line of "AVN 52" Risk	Number	Float number from 0 to 1. (only positive value)
OF_AVN52	Offered Differential of "AVN 52" Risk	Number	Float number from 0 to 1.
FD_AVN52	Final Differential of "AVN 52" Risk	Number	Float number from 0 to 1.
DPA_AVN52	Deposit Premium Allocation of "AVN 52" Risk	Number	Float number from 0 to 1.

Continued on next page

Table 7.1 – continued from previous page

Feature	Feature Description	Type	Value
EXPOSURE_HWMAV	Number that represents the Maximum Agreed Value of the "Hull War Maximum Agreed Value" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
SL_HWMAV	Signed Line of "Hull War Maximum Agreed Value" Risk (valorized in case FLAG_PREMIUM_ _ALLOCATION = 'T')	Number	Float number from 0 to 1. (only positive value)
WL_HWMAV	Writte Line of "Hull War Maximum Agreed Value" Risk	Number	Float number from 0 to 1. (only positive value)
OF_HWMAV	Offered Differential of "Hull War Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
FD_HWMAV	Final Differential of "Hull War Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
DPA_HWMAV	Deposit Premium Allocation of "Hull War Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
E_HULL	Exposure Level of Eastern Fleet - Hull Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
E_PPL	Exposure Level of Eastern Fleet - PPL Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
E_TPL	Exposure Level of Eastern Fleet - TPL Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
W_HULL	Exposure Level of Western Fleet - Hull Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
W_PPL	Exposure Level of Western Fleet - PPL Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
W_TPL	Exposure Level of Western Fleet - TPL Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)

7.3 Initial Analysis of the Dataset

A good place to start analyzing a dataset is to get familiar with the contents and format of the various columns that the dataset contains, as well as the data type of each column.
By examining our datasets through the lens of descriptive statistics, it is possibile to evaluate data in line with the way machine learning algorithms deal with data. Descriptive statistical analysis involves various measures or descriptions that can use to summarize patterns and relationships in data, using "numbers" produced by mathematical calculations, as well as "visualizations" such as graphs or tables that help to reveal significant information in those numbers.
On the following sections are presenting the main steps of this task with some examples while on the "Attached A" of this thesis is report all the details related: complete python code and complete output (data and plots).

7.3.1 Statistical measures

An example of statistical measures is presented on Table 7.2 where the statistical measures related to the feature "GROSS PREMIUM 100" are listed/ summarized.

	GROSS_PREMIUM_100
count	4059.00
mean	7399787.89
std	22168672.32
min	-4684980.52
25%	203703.70
50%	1611892.59
75%	5395392.59
max	444444444.44

Table 7.2: Descriptive statistics of the column GROSS PREMIUM 100

7.3.2 Using visualizations to analyze data

The numbers produced by various statistical measures can provide significant insights, but visual tools such as charts and maps may reveal concepts about our data that are difficult to be seen by using numbers alone.

Analyzing the cross correlations between the features

The heatmap visualization on Figure 7.1 shows correlations between different features in the dataset as numeric values, but enhances them with color coding that helps us identifiying see which values correlate the most.

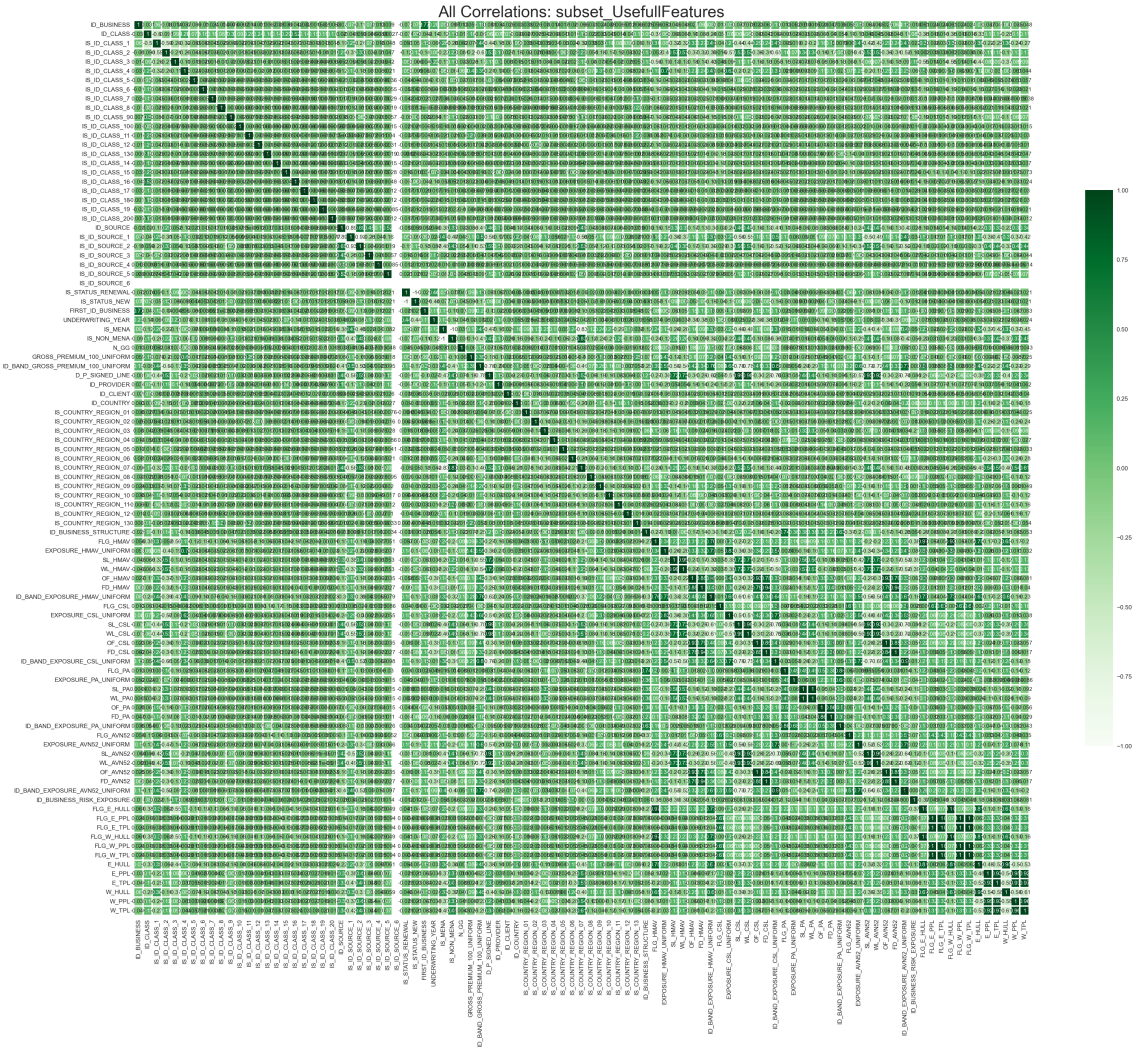

Figure 7.1: Cross correlations between different features in the dataset

- Each feature is shown on the x-axis and y-axis.
- At each intersection, the correlation coefficient is shown for the combination of features represented on the two axes.
- Darker tones highlight values with high correlation coefficient.
- The darkest values appear diagonally where features intersect with themselves.

The heatmap visualization on Figure 7.2 shows cross correlations between different features only related to the features which are extracted from the database (original features without feature engineering).

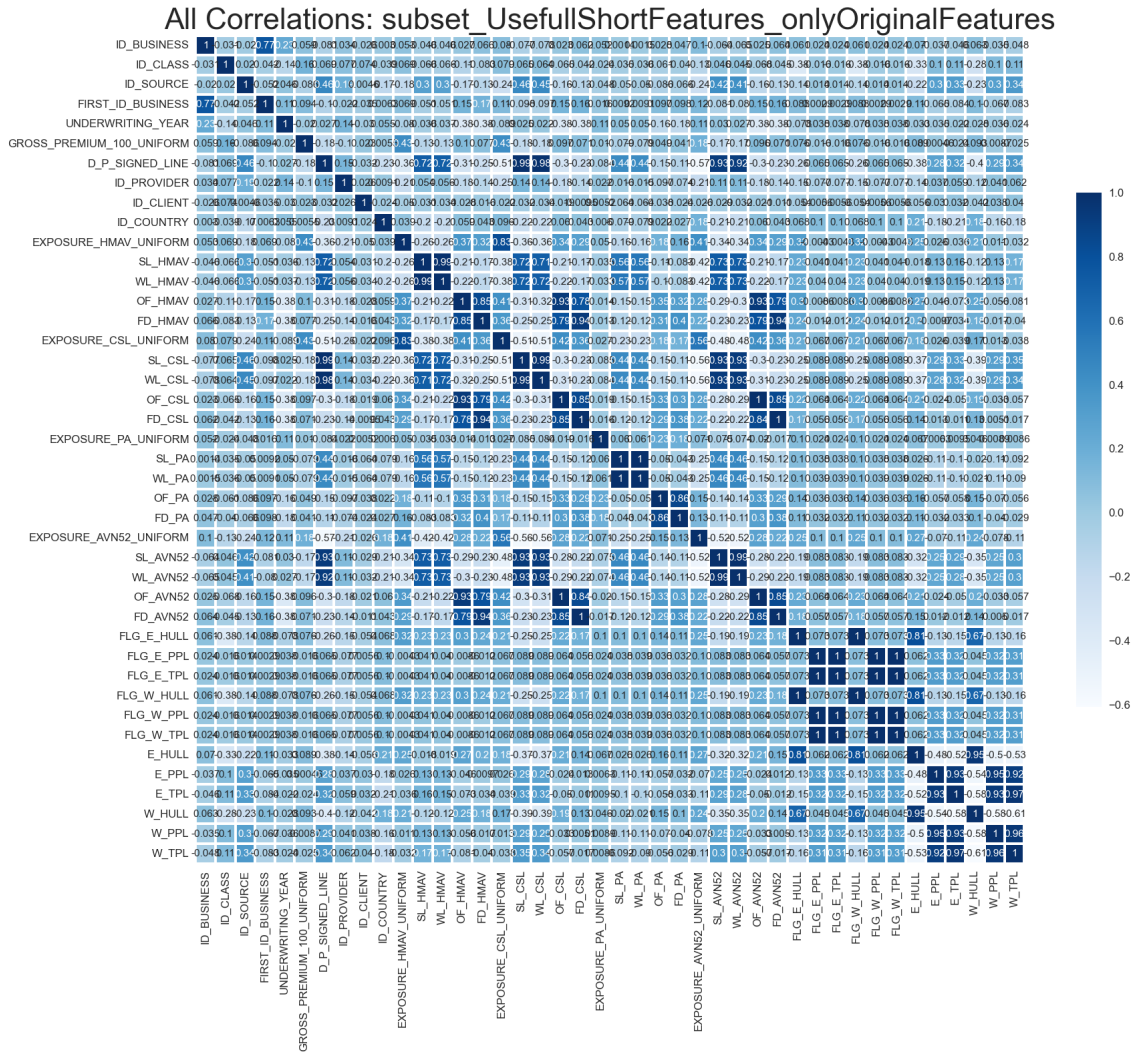

Figure 7.2: Cross correlations between features which are extracted from the database (original features without feature engineering)

Use histograms to visualize the distribution of various features

The histograms on Figure 7.3 uses a column chart to show the distribution of data over a continuous interval or discrete bins/periods. Each bar in a histogram represents the calculated frequency of that values at each interval or bin. Histograms provide a visual summary that can be quickly interpreted to understand where values are concentrated, where the extremes are located and the general skewness of the distribution.

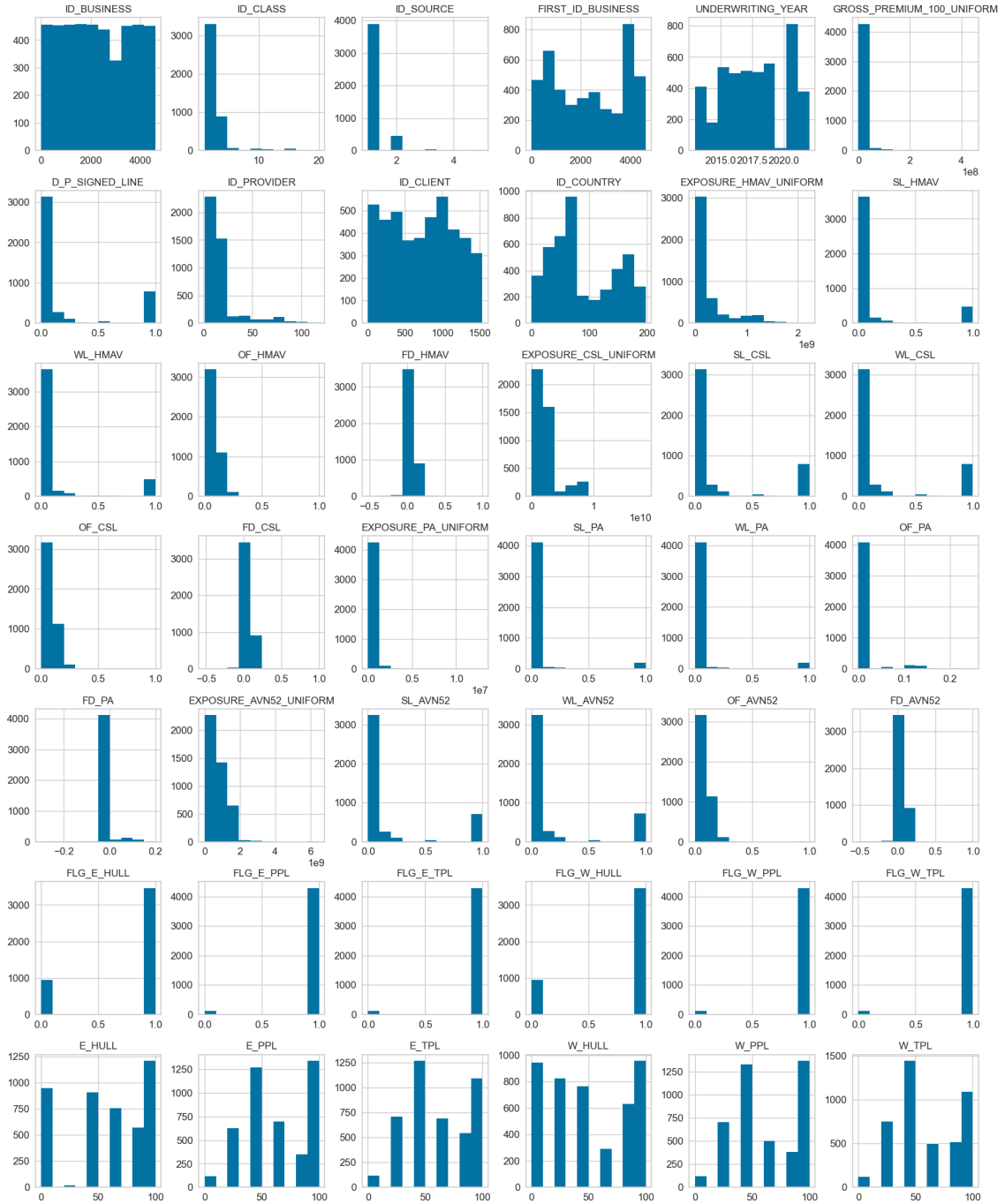

Figure 7.3: Distribution values of the features related to the features directly extracted from the database (original features without feature engineering)

Using Box plot to visualize the distribution of various features

Box plots provide another way of viewing a distribution of data along a line that shows the entire range of values. This type of visualization focuses on showing where data is distributed in relation to the quartiles.

Figure 7.4: Values distribution of the feature GROSS PREMIUM 100

A box plot as on Figure 7.4 can tell us where we have outliers, how symmetrical the distribution is, how tightly data are grouped in the distribution, and how the data is skewed. Before a dataset can be used with a machine learning model, there are typically various tasks to follow up with:

- Data cleaning
- Feature engineering

The initial analysis of the dataset presented will be useful for the next steps.

7.4 Prepare Data of the Dataset: Data cleaning

It is possible to start the data cleaning process once the dataset has been clearly understood by identifying what is included from whatever is missed. This is the process of locating and removing errors and inconsistencies in data. It may involve task such as removing or handling incorrect or missing data, outlier, and so forth. Most part of this task was already performed during the data migration phase described on Chapter 4 with a particular focus on paragraph 4.2.1 *Cleaning and defining data activities*.

Data standardization

A data standardization is necessary in order to perform a correct data cleaning steps like a drop of all outliers and null values. For these reasons, the following new features have been defined as to represent all amounts with the same currency.

- EXPOSURE_AVN52_UNIFORM represents the same amount of the features EXPOSURE_AVN52 but the value is expressed in AED currency.
- EXPOSURE_CSL_UNIFORM represents the same amount of the features EXPOSURE_CSL but the value is expressed in AED currency.
- EXPOSURE_HMAV_UNIFORM represents the same amount of the features EXPOSURE_HMAV but the value is expressed in AED currency.
- EXPOSURE_HWMAV_UNIFORM represents the same amount of the features EXPOSURE_HWMAV but the value is expressed in AED currency.
- EXPOSURE_PA_UNIFORM represents the same amount of the features EXPOSURE_PA but the value is expressed in AED currency.
- GROSS_PREMIUM_100_UNIFORM represents the same amount of the features GROSS_PREMIUM_100 but the value is expressed in AED currency.

Oracle objects created/utilized to perform this task:

- creates the PL/SQL function named GET_EXCHANGE_RATE to extract the correct exchange rate related to the current business. This value will be use to convert the amounts in AED currency.
- creates the PL/SQL functions named PL/SQL GET_UNIFORM_EXPOSURE and GET_UNIFORM_AGROSS_PREMIUM_100 to uniform all the currency amount with the same currency (AED).
- updates the PL/SQL views : V_EXPORT_DATA, V_CSV_DATA and M_V_EXPORT_DATA to integrate the new features.

Data cleaning

As described previously, during the analysis of the dataset it has been examined data values. As an example: looking the box plot about values distribution of the feature GROSS PREMIUM 100 on Figure 7.4 we can observe : very small, big and some negative values. As consequence, all values under 0 AED and over 200000000 AED have been cut.
The result of the cutting samples phase operated after the data standardization and the data analysis phase it is presented here-below where the number of samples after every cutting step have been indicated.
Each filter condition comes from an analysis of the values distribution and a functional analysis made with the business unit insurance team.

4618 Business in the dataset

4605 Business remained after dropping those negative N_GG
—> N_GG>0

4603 Business remained after dropping those "Signed Line" and "Offered Line" over 1
—>(SL_HMAV<=1, OF_HMAV<=1, SL_CSL<=1, OF_CSL<=1, ...)

4602 Business remained after dropping on GROSS_PREMIUM_100 values condition
—> GROSS_PREMIUM_100<= 65.000.000,00

4602 Business remained after dropping on EXPOSURE_HMAV values condition
—> EXPOSURE_HMAV<= 400.000.000,00 USD

4596 Business remained after dropping on EXPOSURE_CSL values condition
—> EXPOSURE_CSL<= 2.500.000.000,00 USD

4588 Business remained after dropping thonose EXPOSURE_PA values condition
—> EXPOSURE_PA<= 1.000.000,00

4579 Business remained after dropping on EXPOSURE_AVN52 values conditions
—> EXPOSURE_AVN52 <= 350.000.000,00 USD if Business Class : MAJOR AIRLINES (ID_CLASS = 4)
—> EXPOSURE_AVN52 <= 500.000.000,00 USD if Business Class : MINOR AIRLINES (ID_CLASS = 1)
—> EXPOSURE_AVN52 <= 750.000.000,00 USD if Business Class : GENERAL AVIATION (ID_CLASS = 2)
—> EXPOSURE_AVN52 <= 350.000.000,00 USD if other Business Classes (ID_CLASS not in (1,2,4))

4408 Business remained after dropping on GROSS_PREMIUM_100 values condition
—> GROSS_PREMIUM_100 NOT NULL
—> GROSS_PREMIUM_100 > 0

Oracle objects created/utilized to perform this task:

- creates the PL/SQL view named V_EXPORT_DATA_REDUCED that contain the same features of the V_EXPORT_DATA view but with the filters just presented.
- creates the PL/SQL view named V_CSV_DATA_REDUCED that contain the same features of the V_CSV_DATA view but with the filters just presented.

7.5 Prepare Data of the Dataset: Feature engineering

After datas are cleaned, consistent and well structured (see the previous paragraph and the data migration phase described in the Chapter 4 with particular focus on paragraph 4.2.1 *Cleaning and defining data activities*), some additional modifications may apply as to prepare them for use in machine learning.

For example, a single column might contain multiple values, such as house number, street name, city and state all in a single value that it must be extracted into separate columns.
In this chapter are presented all the new features created, whilst all features (original features + new features) will be described on paragraph 7.6.

7.5.1 New features coming from the data extraction phase

On the following bullet points are defined the new features created during the extraction data phase.

- FIRST_ID_BUSINESS. This feature manages the information about renewals and assigns the same value for each specific groups of policy (the firs policy and the forward renewals).
  The original feature named PREVIOUS_POLICY_NUMBER cannot be used by an AI algorith because does not aggregate data of the same business (the first policy and the forward renewals), but provides just information about the previous and next policy, whilst the new one can do that.

- The value of the feature N_GG does not occurin the database, but it corresponds to the result of an Oracle's function realized to determine the number of days between the INCEPTION and EXPIRY dates.
  The original features (INCEPTION and EXPIRY dates) cannot be used by an AI algorithm because provide just an information about dates and for our goals a specific inception/ expiry policy dates are not significant whilst the length of validity period of the policy is.

- All features with the name starting with FLG_... are features not presented in the database and corrisponde to a new boolean features as to indicate the presence of a specific sub-set of information for a specific record, if any.
  For instance, a record with the feature FLG_CLS equals to 1/true means the presence of a Combined Single Limit Risk for the specific policy then the valorization of the following features is expected: EXPOSURE_CSL, SL_CSL, WL_CSL, OF_CSL, FD_CSL (see the meanings on table 7.1)

- All the features with the name starting with IS_... are new features which are not inculded in the database. These features are used to convert a single

feature with enumarated values to a set of features with boolean values. For example the feature ID_SOURCE with 6 enumerated possible values had let to create 6 new feature IS_ID_SOURCE_1, IS_ID_SOURCE_2, ..., IS_ID_SOURCE_6,. The feature IS_ID_SOURCE_n with value 1/true means that the feature ID_SOURCE assume value n.

Oracle objects created/utilized to perform this task:

- creates PL/SQL function to determine the new features values.
- updates the PL/SQL views : V_EXPORT_DATA, V_CSV_DATA and M_V_EXPORT_DATA to integrate the new features.

7.5.2 New features coming from the data normalization phase

All currency amounts are been normalize to rappresent all the value with the same currency AED.

As already described in the previous chapter , this was elaborated with an Oracle's function realized to manage the currencies conversion by using the rate changes included in the database.

The new features are:

- EXPOSURE_AVN52_UNIFORM
- EXPOSURE_CSL_UNIFORM
- EXPOSURE_HMAV_UNIFORM
- EXPOSURE_HWMAV_UNIFORM
- EXPOSURE_PA_UNIFORM
- GROSS_PREMIUM_100_UNIFORM

7.5.3 New feature coming from the Analysis of Business Structure

Starting from the idea that the combination of the features:

"FLG_HMAV" , "FLG_CSL" , "FLG_PA" , "FLG_AVN52" , "FLG_HWMAV"

as to represents a type of complex risk with a combination of different types of basic risks;it has been created a new feature named "ID_BUSINESS_STRUCTURE". This feature aggregates the information of five feature in ones.

Oracle objects created/utilized to perform this task:

- creates table: T_SUP_BUSINESS_STRUCTURE with primary key the column named ID_BUSINESS_STRUCTURE. This table contains all the combination of the features: "FLG_HMAV", "FLG_CSL", "FLG_PA", "FLG_AVN52" and "FLG_HWMAV".
- updates the PL/SQL views : V_EXPORT_DATA, V_CSV_DATA and M_V_EXPORT_DATA to integrate the new feature.

7.5.4 New feature coming from the Analysis of the Business Risk Structure

Starting from the idea that the combination of the features:

```
"FLG_E_HULL", "FLG_E_PPL", "FLG_E_TPL",
"FLG_W_HULL", "FLG_W_PPL", "FLG_W_TPL"
"E_HULL", "E_PPL", "E_TPL",
"W_HULL", "W_PPL", "W_TPL"
```

represents a type of complex exposure risk as a combination of different type of exposure basic risk it has been created a new feature named "ID_BUSINESS_EXPOSURE". This feature aggregates the information of 12 features in one.

Oracle objects created/utilized to perform this task:

- creates table: T_SUP_BUSINESS_EXPOSURE with primary key the column named ID_BUSINESS_EXPOSURE. This table contains all the combination of the features: "FLG_E_HULL", "FLG_E_PPL", "FLG_E_TPL", "FLG_W_HULL", "FLG_W_PPL", "FLG_W_TPL", "E_HULL", "E_PPL", "E_TPL", "W_HULL", "W_PPL" and "W_TPL".
- updates the PL/SQL views : V_EXPORT_DATA, V_CSV_DATA and M_V_EXPORT_DATA to integrate the new feature.

7.5.5 New features coming from the features analysis related both to Risk Exposures and Gross Premium 100

During the initial analysis of the dataset (described on chapter 7.3) and the sessions analysis made with the business unit it was developped the idea that for our goals, it is irrilevant a specific amount value, but the membership of this value to a specific value band. This concept can be applied to all features related to currency amounts. For example: if we have 3 policies with GROSS_PREMIUM_100 feature values equal to "12.000.000,00", "1.00.000.000,00" and "12.005.00,00"; for our goals it is not significant to distinguish 3 different values of GROSS_PREMIUM_100 whilst it is mandatory to distinguish 2 groups values "Group values A" which contain the value 1.00.000.000,00 and "Group values B" which contain 12.000.000,00 and 12.005.00,00 values.

Analysis

First of all it is has been analyzed the frequency distribution of the feature "Gross Premium 100" (see on Figure 7.5) and the frequency distribution related to the risk exposure features (see on Figures: 7.6, 7.7, 7.7, 7.8 and 7.9)

Oracle objects created/utilized to perform this task:

- creates PL/SQL view: V_VALUE_BAND_ANALYSIS to analyze the value frequencies related to the risk exposure features (AVN52, CSL, HMAV, PA, HWMAV) and to the feature "Gross Premium 100" with the amounts expressed in AED currency.
- creates PL/SQL view: V_VALUE_BAND_ANALYSIS_USD which correspond to the V_VALUE_BAND_ANALYSIS view but with the values expressed in USD currency.

After the detailed analysis, 3 different values grouping technique have been tested. These techniques which are summarized int the following list, will be presented and compared later.

- Sturges rule
- Percentage frequency distribution rule
- Pick frequency distribution rule

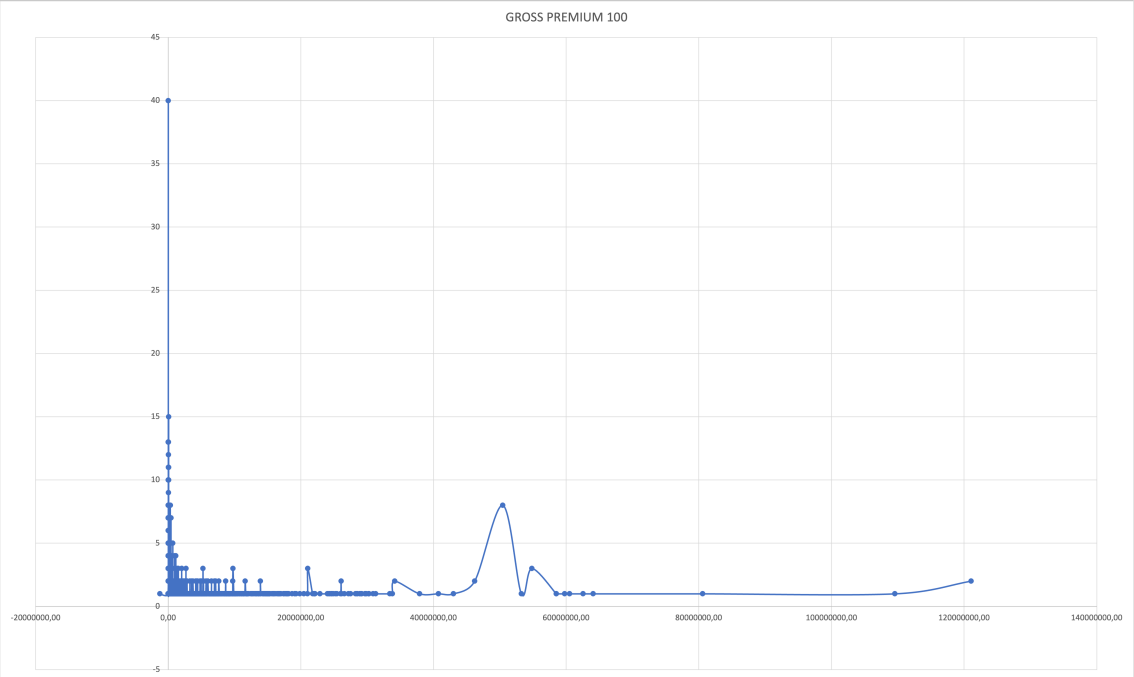

Figure 7.5: Frequency distribution of the feature : GROSS_PREMIUM_100

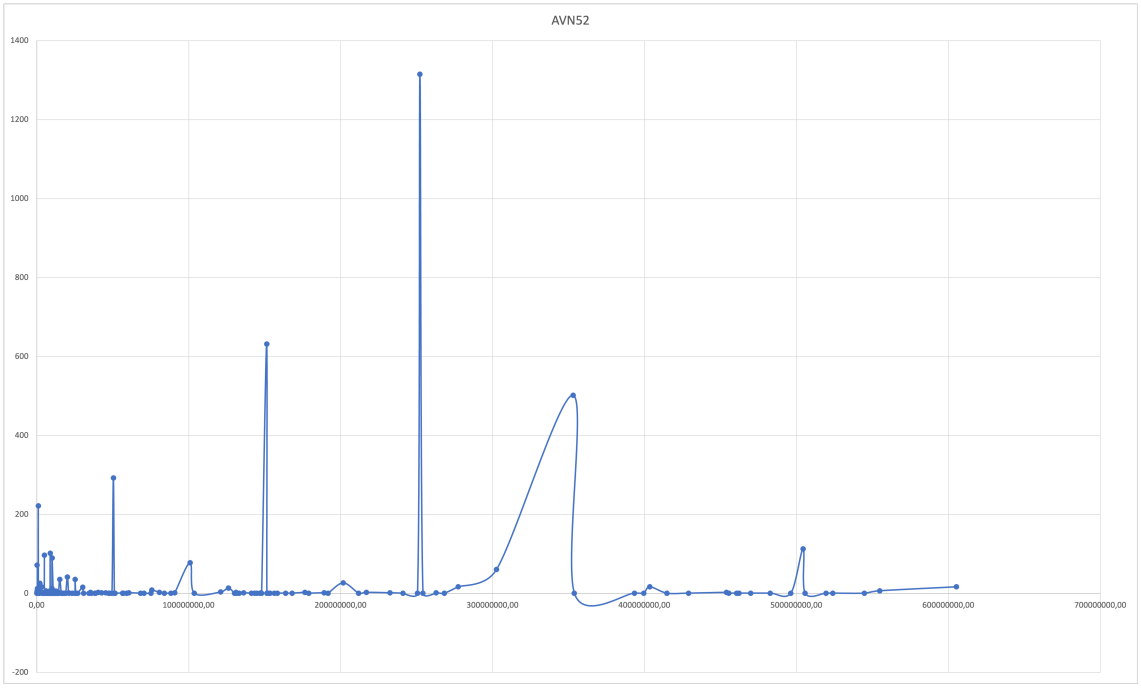

Figure 7.6: Frequency distribution of the feature : AVN52

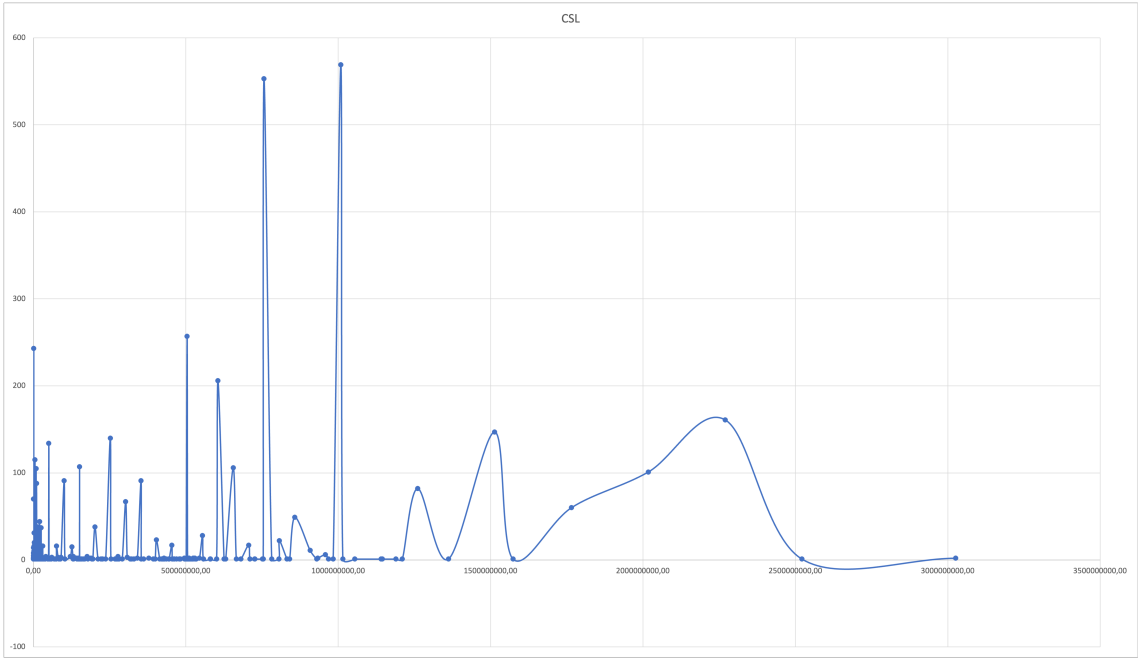

Figure 7.7: Frequency distribution of the feature : CSL

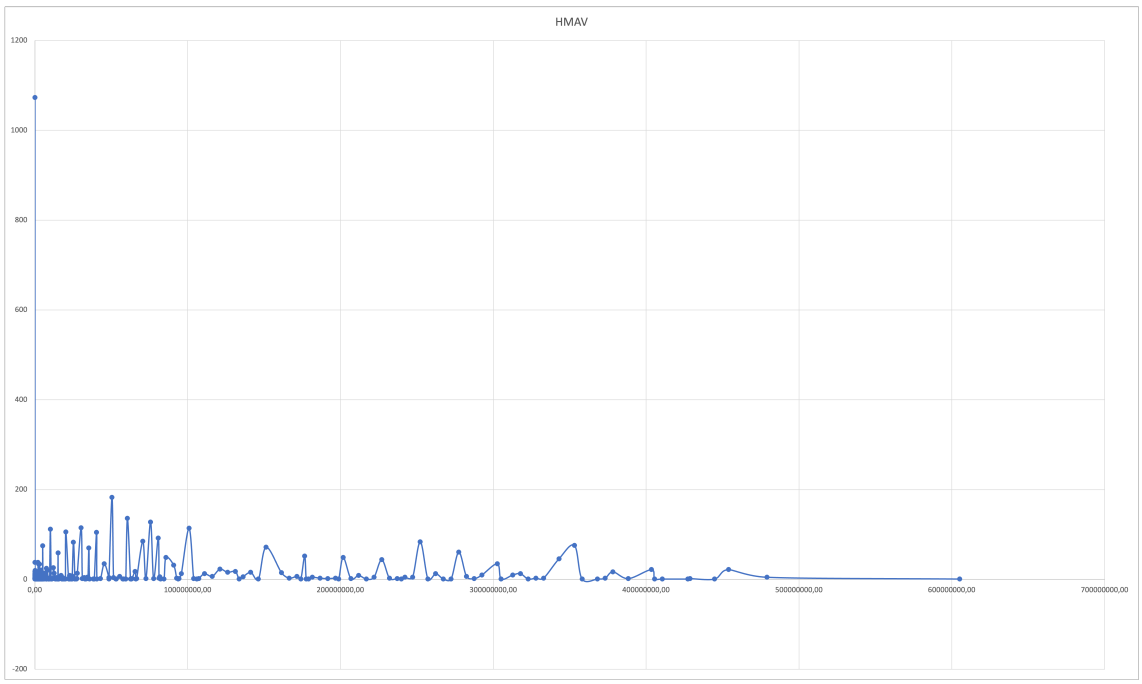

Figure 7.8: Frequency distribution of the feature : HMAV

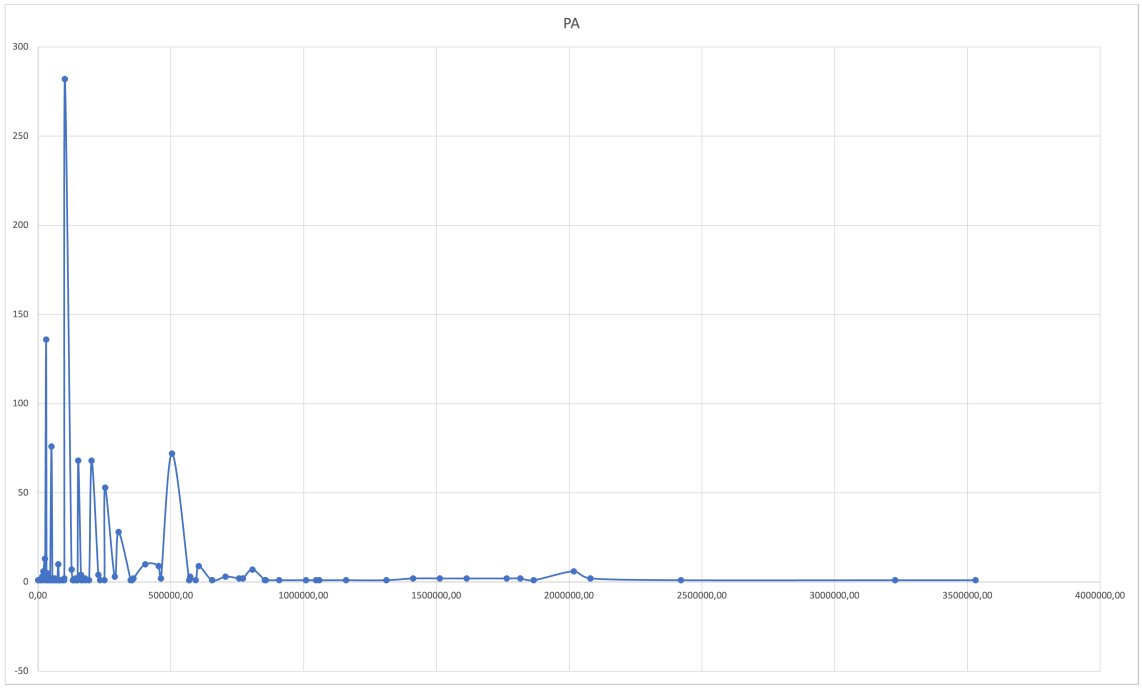

Figure 7.9: Frequency distribution of the feature : PA

Sturges rule

Sturges rule is a rule for determining the desirable number of groups into which a distribution of observations should be classified; defined N as the number of observations then the number of groups classes K is determinate as follow.
In our case N corresponds to the number of different values assumed by a specific feature whilst K corresponds to the number of value bands created to grouping the different values assumed by a specific feature.

$$K = 1 + \frac{10}{3}\log_{10}(N)$$

The rule application example presented in Figure 7.10 shows how the yellow lines, which define the groups, create K values bands all with the same length.

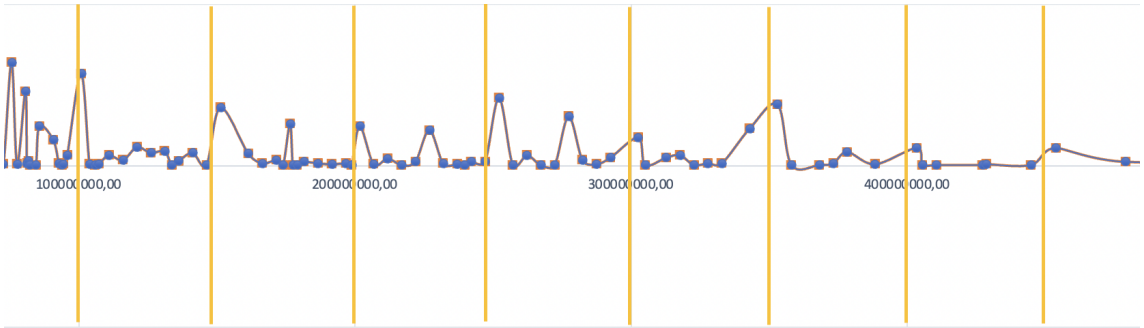

Figure 7.10: Example application of Sturges rule

Percentage frequency distribution rule

Percentage frequency distribution rule is a rule to define a set of classes/groups/bands/intervals which assure the same number of different value for each class/group/band/interval.
Defining:

- I_{C_i} = Interval of Values related to i_{th} Class
- V_k = Value k_{th} assumed by our data
- f_{V_k} = Frequency of a value k_{th}

the number of values been grouping in the same class/group/band/interval I_{C_i} is equals to

$$f_{TOT_{C_i}} = \sum_{V_k \in I_{C_i}} f_{V_k}$$

Defining the percentage of different values grouping in the same class/group/band/interval C_i compared to the number of different values assumed by the data as follow:

$$Percentage_{C_i} = \frac{f_{TOT_{C_i}}}{\sum_{i=1}^{w} f_{TOT_{C_i}}}$$

the procedure to define a set of classes/groups/bands/intervals is:

- chooses a minimal Percentage of frequency Q
- defines W different classes/groups/bands/intervals assuring that every class C_i have $Percentage_{C_i} >= Q$

The rule application example presented in Figure 7.11 shows how the yellow lines, which define the groups, create K values bands with different length.

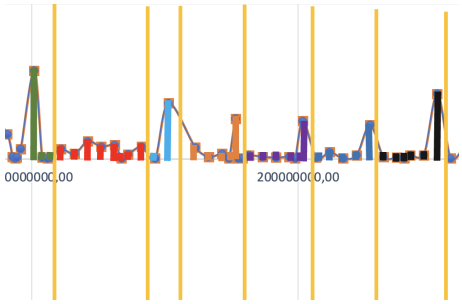

Figure 7.11: Example application of Percentage frequency distribution rule

Looking at the example on Figure 7.12, defining classes/groups/bands/intervals with this rule means in our case that business related to values colored in red will be considered as "same type of business" and so on...

This type of value grouping comes from the "Percentage frequency distribution rule" which has been discussed with the business unit. The conclusion of this confrontation phase shows that this rule is not applicable in our case necessarily, since it can occur that a type of business with low frequency (for example a specific type of policy) might be bring back to a different type of business (type policy with hight frequency).

Looking at the example on Figure 7.12, the policy with assumed value A should be bring back to the policy with value C but with this rule is bring back to the value B.

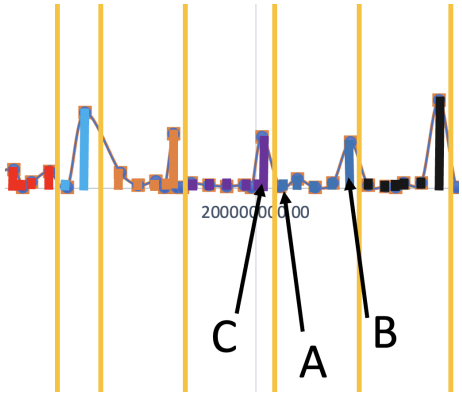

Figure 7.12: Example error application of Percentage frequency distribution rule

Peak frequency distribution rule

Peak frequency distribution rule is a rule as to define a set of bands/intervals with the scope of bringing back a value to the nearest more frequent value.

This idea comes from a conversation with the business unit. Such an assumption may be good to get our target because "the system" in this way:

- should consider the values with a highest number of policies related to specific type of policy/business;
- should bring back a particolar value to the closest type of policy/business;

As an example: a policy/business with the GROSS_PREMIUM_100 feature which equals to "12.000.231,32" will be consider as equal to the type of policy/business with the GROSS_PREMIUM_100 feature equal to "12.000.000,00".

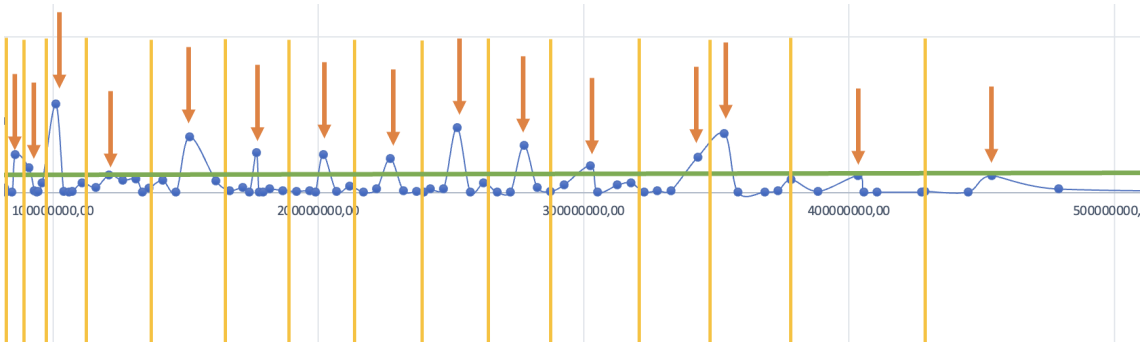

Figure 7.13: Example application of Peak frequency distribution rule

The rule application example presented in Figure 7.13 shows how the yellow lines, which define these groups, create K values bands with different length, every bands is related to a frequency distribution value peak and the green line defines the threshold used to define a value both as a peak or not.

In order to identify the peaks it has been defined the following percentage that helps to set a threshold expressed with a percentage value and not an absolute value.

$$\% = \frac{(\text{"}\#ofDifferentValues\text{"} - \text{"}\#ofpeaks\text{"})}{\text{"}\#ofDifferentValues\text{"}}$$

Oracle objects created/utilized to perform this task:

- creates the table : T_SUP_PERCENT_PEAK_THRESHOLD to define different percentage treshold value levels to be test for a specific feature
- creates PL/SQL view : V_PEAK_IDENTIFIED to identify the peaks for a specific feature using the percentage treshold value defined on table T_SUP_PERCENT_PEAK_THRESHOLD

On Table 7.3 are summarized the analysis made for each and every specific features related both to Risk Exposures and Gross Premium 100.

FEATURES (without null values)	**AVN52**	**CSL**	**HMAV**	**PA**	**GROSS_PREMIUM_100**
# of records (not null)	4182	4308	3502	973	4408
# of Values	218	288	435	101	3669
Sturges Rule (K)	13.1 (-96.0%)	13.1 (-96.8%)	12.8 (-97.7%)	11.0 (-92.4%)	13.1 (-99.6%)
# of peaks (f >0.01%)	218 (.0%)	288 (.0%)	435 (0.0%)	101 (0.0%)	3669 (.0%)
# of peaks (f >0.02%)	218 (.0%)	288 (.0%)	435 (0.0%)	101 (0.0%)	3669 (.0%)
# of peaks (f >0.023%)	218 (.0%)	288 (.0%)	435 (0.0%)	101 (0.0%)	355 (-90.3%)
# of peaks (f >0.025%)	79 (-63.8%)	102 (-64.6%)	435 (0.0%)	101 (0.0%)	355 (-90.3%)
# of peaks (f >0.035%)	**79 (-63.8%)**	**102 (-64.6%)**	**202 (-53.6%)**	101 (0.0%)	**355 (-90.3%)**
# of peaks (f >0.05%)	60 (-72.5%)	72 (-75.0%)	202 (-53.6%)	101 (0.0%)	127 (-96.5%)
# of peaks (f >0.1%)	39 (-82.1%)	50 (-82.6%)	119 (-72.6%)	101 (0.0%)	43 (-98.8%)
# of peaks (f >0.5%)	18 (-91.7%)	31 (-89.2%)	43 (-90.1%)	**19 (-81.2%)**	1 (-100.0%)
# of peaks (f >1%)	14 (-93.6%)	24 (-91.7%)	26 (-94.0%)	12 (-88.1%)	0 (-100.0%)

Table 7.3: Peak and Bands identified analysis summary

Observing the Table 7.3 :

- each column corresponds to a specific feature
- the row "# of records (not null)" indicates the number of not null values included in our database for the specific feature;
- the row "# of Value" indicates the number of different value amounts for the specific feature;
- the row "Sturges Rule" indicates the number of classes/groups/bands/intervals for the specific feature determinated by the Sturges Rule.
- the rows "# of peaks (f> **%)" indicates the number of peaks identified for a specific feature and the specific percentage expressed and equals to **. At every peak corresponds a specific class/group/band/interval;
- the percentage value expressed under parentheses in the rows "# of peaks (f> **%)" indicates the decrease percentage of values as a ratio between "# of Value" and the number of classes/groups/bands/intervals defined for the specific feature;
- the cells highlighted in orange indicates the number of classes/groups/bands/intervals defined for each feature.

As an example, analyzing the feature named "GROSS_PREMIUM_100":

- has 4408 values in the database
- has 3669 different values in the database
- it defines 355 different bands (see the cell highlighted in orange) which means that a decrease of the possible values from 3669 to 355 equals to -90.3%.

After the number of bands for each feature has been selected (see the cells highlighted in orange in the Table 7.3), it has been also defined that the following new features indicate the ID of the band where every each value is comprised.

- ID_BAND_EXPOSURE_AVN52_UNIFORM : ID of the band where each and every value of the EXPOSURE_AVN52_UNIFORM feature is comprised.
- ID_BAND_EXPOSURE_CSL_UNIFORM : ID of the band where each and every value of the EXPOSURE_CSL_UNIFORM feature is comprised.
- ID_BAND_EXPOSURE_HMAV_UNIFORM : ID of the band where each and every value of the EXPOSURE_HMAV_UNIFORM feature is comprised.
- ID_BAND_EXPOSURE_HWMAV_UNIFORM : ID of the band where each and every value of the EXPOSURE_HWMAV_UNIFORM feature is comprised.
- ID_BAND_EXPOSURE_PA_UNIFORM : ID of the band where each and every value of the EXPOSURE_PA_UNIFORM feature is comprised.
- ID_BAND_GROSS_PREMIUM_100_UNIFORM : ID of the band where each and every value of the GROSS_PREMIUM_100_UNIFORM feature is comprised.

Oracle objects created/utilized to perform this task:

- creates the PL/SQL view : V_PEAK_SELECTED to identify all peaks taken after the selection of the number of bands for each feature (see the cells highlighted on orange on the table)
- creates the table T_SUP_PEAK_SELECTED which contains all peaks selected from the view V_PEAK_SELECTED
- creates the table T_SUP_VALUE_BAND which contains all bands for each and every feature
- creates the PL/SQL procedure : DEFINE_BAND as to populate the table T_SUP_VALUE_BAND starting from the table T_SUP_PEAK_SELECTED. The first and the last value of each and every band corresponds to the average value between two peaks.
- creates the PL/SQL function GET_ID_VALUE_BAND which has two inputs: the value of the feature and the name of the feature. This fuction returns the ID of the band to which the value belongs. This function uses the table T_SUP_VALUE_BAND.
- updates the PL/SQL views : V_EXPORT_DATA, V_CSV_DATA and M_V_EXPORT_DATA to integrate the new features.

The following Figures 7.14, 7.15, 7.16, 7.17 and 7.18 represent the distribution of different values for a specific feature through the various bands defined. This can be compared with the Figures: 7.6, 7.7, 7.7, 7.8 and 7.9).

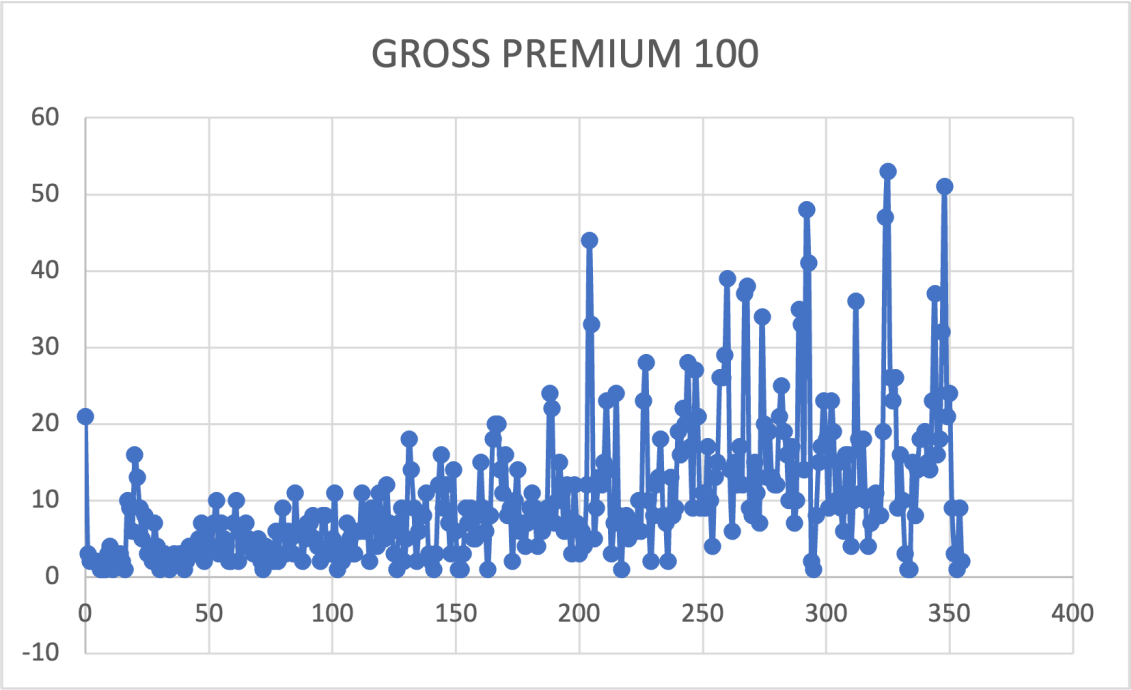

Figure 7.14: GROSS_PREMIUM_100 feature's values distribution on identified bands. (Values not records)

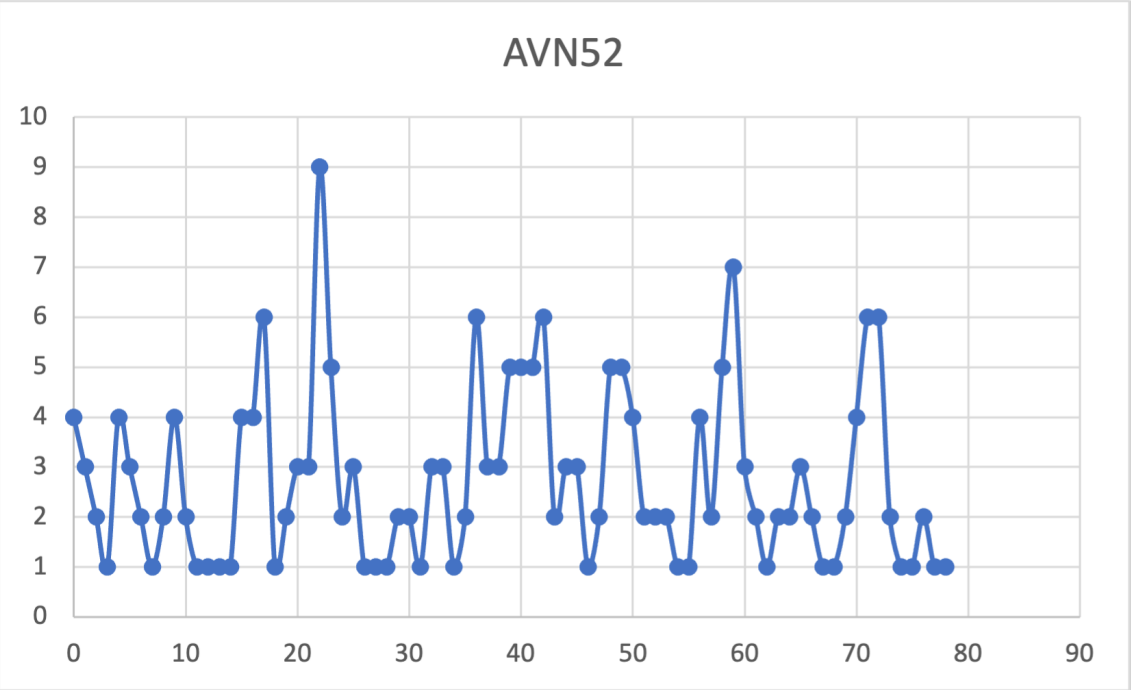

Figure 7.15: AVN52 feature's values distribution on identified bands. (Values not records)

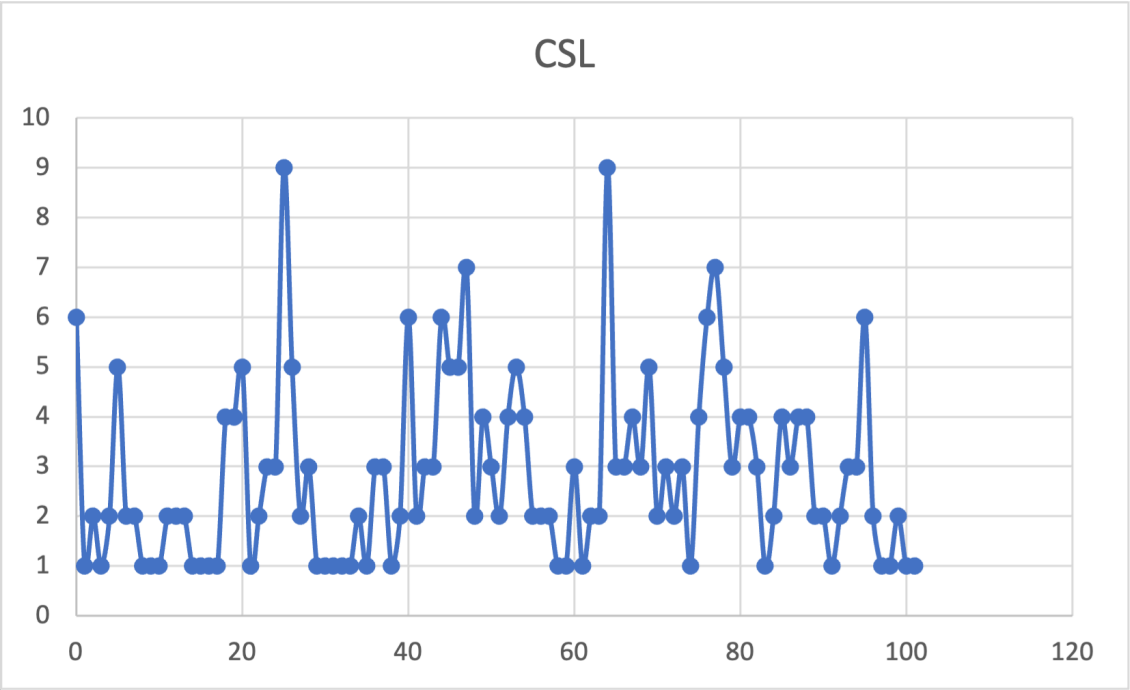

Figure 7.16: CSL feature's values distribution on identified bands. (Values not records)

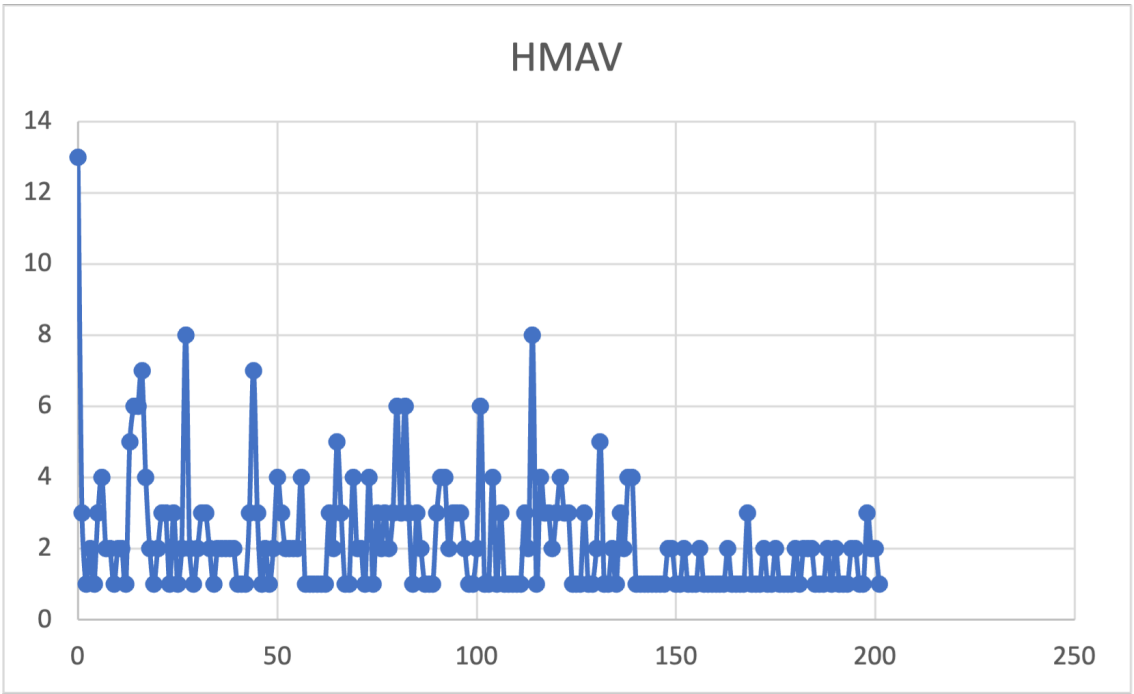

Figure 7.17: HAMV feature's values distribution on identified bands. (Values not records)

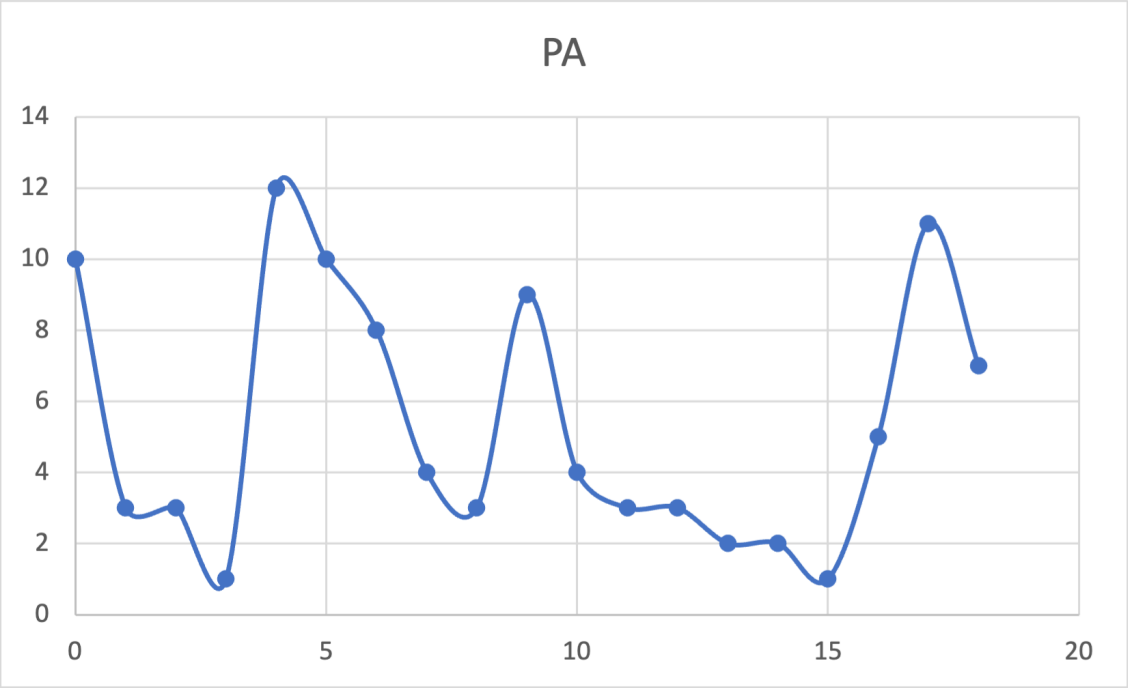

Figure 7.18: PA feature's values distribution on identified bands. (Values not records)

7.6 Dataset description

The following Table 7.4 describes the dataset which is used to build all the models discussed in this thesis. The dataset is extracted from the database by using different oracle views and procedures as to aggregate and format data in a useful format. This dataset includes both the original characteristics and the new ones defined in the previous paragraphs.

Table 7.4: Dataset extracted from database

Feature	Feature Description	Type	Value
ID_BUSINESS	Identifies in a unique way a Business	Number	ID
COD_PHASE	Identifies the business phase	String	P : Pre-Underwriting - U : Underwriting - B : Booking - E : Endorsement
BUSINESS_STATUS	Business status	String	C : Close - O : Open
ID_CLASS	Business Class type	Number	ID
CLASS		String	Decode of ID_CLASS from table T_CLASS
IS_ID_CLASS_1		Number	1 : ID_CLASS = 1 (MINOR AIRLINES) else 0
IS_ID_CLASS_2		Number	1 : ID_CLASS = 2 (GENERAL AVIATION) else 0
Continued on next page			

Table 7.4 – continued from previous page

Feature	Feature Description	Type	Value
IS_ID_CLASS_3		Number	1 : ID_CLASS = 3 (MISCELLANEOUS LIABILITY) else 0
IS_ID_CLASS_4		Number	1 : ID_CLASS = 4 (MAJOR AIRLINES) else 0
IS_ID_CLASS_5		Number	1 : ID_CLASS = 5 (CONTRACTORS LIABILITY) else 0
IS_ID_CLASS_6		Number	1 : ID_CLASS = 6 (AIRPORT LIABILITY) else 0
IS_ID_CLASS_7		Number	1 : ID_CLASS = 7 (ATC LIABILITY) else 0
IS_ID_CLASS_8		Number	1 : ID_CLASS = 8 (XS LIABILITY) else 0
IS_ID_CLASS_9		Number	1 : ID_CLASS = 9 (CONTIGENCY LIABILITY) else 0
IS_ID_CLASS_10		Number	1 : ID_CLASS = 10 (PREMISES LIABILITY) else 0
IS_ID_CLASS_11		Number	1 : ID_CLASS = 11 (PRODUCTS LIABILITY) else 0
IS_ID_CLASS_12		Number	1 : ID_CLASS = 12 (REFUELLING LIABILITY) else 0
IS_ID_CLASS_13		Number	1 : ID_CLASS = 13 (LOSS OF LICENSE) else 0
IS_ID_CLASS_14		Number	1 : ID_CLASS = 14 (PERSONAL ACCIDENT) else 0
IS_ID_CLASS_15		Number	1 : ID_CLASS = 15 (INWARD TREATY) else 0
IS_ID_CLASS_16		Number	1 : ID_CLASS = 16 (LIABILITY) else 0
IS_ID_CLASS_17		Number	1 : ID_CLASS = 17 (MRO LIABILITY) else 0
IS_ID_CLASS_18		Number	1 : ID_CLASS = 18 (XS PRODUCT LIABILITY) else 0
IS_ID_CLASS_19		Number	1 : ID_CLASS = 19 (AIRMEET LIABILITY) else 0
IS_ID_CLASS_20		Number	1 : ID_CLASS = 20 (XS HULL AND LIABILITY) else 0
ID_SOURCE	Business Source type	Number	ID
SOURCE		String	Decode of ID_SOURCE from table T_SOURCE
IS_ID_SOURCE_1		Number	1 : ID_SOURCE= 1 (INWARD FACULTATIVE) else 0

Continued on next page

Table 7.4 – continued from previous page

Feature	Feature Description	Type	Value
IS_ID_SOURCE_2		Number	1 : ID_SOURCE= 2 (DIRECT) else 0
IS_ID_SOURCE_3		Number	1 : ID_SOURCE= 3 (FRONTING AND SERVICING) else 0
IS_ID_SOURCE_4		Number	1 : ID_SOURCE= 4 (CANCELLED) else 0
IS_ID_SOURCE_5		Number	1 : ID_SOURCE= 5 (BROKERAGE) else 0
IS_ID_SOURCE_6		Number	1 : ID_SOURCE= 6 (INWARD TREATIES) else 0
STATUS	Business status	String	N : New - R : Renewal
IS_STATUS_RENEWAL		Number	1 : STATUS= 'R' else 0
IS_STATUS_NEW		Number	1 : STATUS= 'N' else 0
PREVIOUS_POLICY_NUMBER	PREVIOUS POLICY (In case STATUS = R)	String	Previous policy number (valorized only if STATUS + 'R')
FIRST_ID_BUSINESS		Number	ID_BUSINESS, Reference on table T_BUSINESS
UNDERWRITING_YEAR	Underwriting Year	Number	Year
MENA_NON_MENA	Mena/Non Mena	String	N : NON MENA - M : MENA
IS_MENA		Number	1 : MENA_NON_MENA= 'M' else 0
IS_NON_MENA		Number	1 : MENA_NON_MENA= 'N' else 0
BUSINESS_COD_CURRENCY	Business Currency type (see decode on table T_CURRENCY)	String	
BUSINESS_EXCHANGE_RATE	Business Currency Exchange Rate (see decode on table T_CURRENCY)	Number	
INCEPTION		Date	Data of Inception (format dd/mm/yyyy)
EXPIRY		Date	Data of Expiry (format dd/mm/yyyy)
N_GG		Number	Number of days between Data of Inception and Data of Expiry
GROSS_PREMIUM_100	Gross Premium 100%	Number	Float number. (The value is expressed in the currency of the business: BUSINESS_COD_CURRENCY)
GROSS_ _PREMIUM_100_UNIFORM	Gross Premium 100%	Number	Float number. (The value is expressed in AED currency)
ID_BAND_GROSS_ _PREMIUM_100_UNIFORM	Gross Premium 100 discretized	Number	ID
POLICY_NUMBER	Policy number	String	

Continued on next page

Table 7.4 – continued from previous page

Feature	Feature Description	Type	Value
FLAG_PREMIUM_ALLOCATION	FLAG Premium Allocation presence	String	T : Presence of Premium Allocation - F : Absence of Premium Allocation
D_P_SIGNED_LINE	Deposit Premium Signed Line	Number	(valorized in case of FLAG_PREMIUM_ _ALLOCATION = 'F')
ID_PROVIDER	Business Provider (see decode on table T_PROVIDER)	Number	ID
PROVIDER		String	Decode of ID_PROVIDER from table T_PROVIDER
ID_CLIENT	Business Client (see decode on table T_CLIENT)	Number	ID
CLIENT		String	Decode of ID_CLIENT from table T_CLIENT
ID_COUNTRY	Business Country (see decode on table T_COUNTRY)	Number	ID
COUNTRY		String	Decode of ID_COUNTRY from table T_COUNTRY
COUNTRY_REGION	Region of Country	String	Country Region from decode table T_COUNTRY
IS_COUNTRY_REGION_01		Number	1 : ID_COUNTRY _REGION= 'Australasia' else 0
IS_COUNTRY_REGION_02		Number	1 : ID_COUNTRY _REGION= 'South Asia' else 0
IS_COUNTRY_REGION_03		Number	1 : ID_COUNTRY_REGION= 'Western Europe' else 0
IS_COUNTRY_REGION_04		Number	1 : ID_COUNTRY_REGION= 'Caribbean' else 0
IS_COUNTRY_REGION_05		Number	1 : ID_COUNTRY_REGION= 'Central America' else 0
IS_COUNTRY_REGION_06		Number	1 : ID_COUNTRY_REGION= 'Africa' else 0
IS_COUNTRY_REGION_07		Number	1 : ID_COUNTRY_REGION= 'Middle East' else 0
IS_COUNTRY_REGION_08		Number	1 : ID_COUNTRY_REGION= 'South America' else 0
IS_COUNTRY_REGION_09		Number	1 : ID_COUNTRY_REGION= 'Far East Asia' else 0
IS_COUNTRY_REGION_10		Number	1 : ID_COUNTRY_REGION= 'Eastern Europe' else 0
IS_COUNTRY_REGION_11		Number	1 : ID_COUNTRY_REGION= 'North Africa' else 0
IS_COUNTRY_REGION_12		Number	1 : ID_COUNTRY_REGION= 'Central Asia' else 0
IS_COUNTRY_REGION_13		Number	1 : ID_COUNTRY_REGION= 'North America' else 0

Continued on next page

Table 7.4 – continued from previous page

Feature	Feature Description	Type	Value
RISK_DOMICILE_ _COD_CURRENCY	Risk domicile Currency type (see decode on table T_CURRENCY)	String	
RISK_DOMICILE_ _EXCHANGE_RATE	Risk domicile Exchange Rate (see decode on table T_CURRENCY)	Number	
FLG_HMAV	FLAG "Hull Maximum Agreed Value" Risk presence	Number	1 : Presence of this risk type - 0 : Absence of this risk type
EXPOSURE_HMAV	Number that represents the Maximum Agreed Value of the "Hull Maximum Agreed Value" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
EXPOSURE_HMAV_UNIFORM	Same as EXPOSURE_HMAV %	Number	Float number. (The value is expressed in AED currency)
ID_BAND_ _EXPOSURE_HMAV_UNIFORM	Same as EXPOSURE_HMAV but the value is discretized	Number	ID
SL_HMAV	Signed Line of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1. (only positive value)
WL_HMAV	Writte Line of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1. (only positive value)
OF_HMAV	Offered Differential of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
FD_HMAV	Final Differential of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
DPA_HMAV	Deposit Premium Allocation of "Hull Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
FLG_CSL	FLAG "Combined Single Limit" Risk presence	Number	1 : Presence of this risk type - 0 : Absence of this risk type
EXPOSURE_CSL	Number that represents the Limit of "Combined Single Limit" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
EXPOSURE_CSL_UNIFORM	Same as EXPOSURE_CSL %	Number	Float number. (The value is expressed in AED currency)
ID_BAND_ _EXPOSURE_CSL_UNIFORM	Same as EXPOSURE_CSL but the value is discretized	Number	ID
SL_CSL	Signed Line of "Combined Single Limit" Risk (valorized in case FLAG_PREMIUM_ _ALLOCATION = 'T')	Number	Float number from 0 to 1. (only positive value)
WL_CSL	Writte Line of "Combined Single Limit" Risk	Number	Float number from 0 to 1. (only positive value)
OF_CSL	Offered Differential of "Combined Single Limit" Risk	Number	Float number from 0 to 1.

Continued on next page

Table 7.4 – continued from previous page

Feature	Feature Description	Type	Value
FD_CSL	Final Differential of "Combined Single Limit" Risk	Number	Float number from 0 to 1.
DPA_CSL	Deposit Premium Allocation of "Combined Single Limit" Risk	Number	Float number from 0 to 1.
FLG_PA	FLAG "Personal Accidents" Risk presence	Number	1 : Presence of this risk type - 0 : Absence of this risk type
EXPOSURE_PA	Number that represents the Sum Insured of "Personal Accidents" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
EXPOSURE_PA_UNIFORM	Same as EXPOSURE_PA %	Number	Float number. (The value is expressed in AED currency)
ID_BAND_EXPOSURE_ _PA_UNIFORM	Same as EXPOSURE_PA but the value is discretized	Number	ID
SL_PA	Signed Line of "Personal Accidents" Risk (valorized in case FLAG_PREMIUM_ _ALLOCATION = 'T')	Number	Float number from 0 to 1. (only positive value)
WL_PA	Writte Line of "personal Accidents" Risk	Number	Float number from 0 to 1. (only positive value)
OF_PA	Offered Differential of "personal Accidents" Risk	Number	Float number from 0 to 1.
FD_PA	Final Differential of "personal Accidents" Risk	Number	Float number from 0 to 1.
DPA_PA	Deposit Premium Allocation of "Personal Accidents" Risk	Number	Float number from 0 to 1.
FLG_AVN52	FLAG "AVN 52" Risk presence	Number	1 : Presence of this risk type - 0 : Absence of this risk type
EXPOSURE_AVN52	Number that represents the Sub Limit of "AVN 52" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
EXPOSURE_AVN52_UNIFORM	Same as EXPOSURE_AVN52 %	Number	Float number. (The value is expressed in AED currency)
ID_BAND_EXPOSURE_ _AVN52_UNIFORM	Same as EXPOSURE_AVN52 but the value is discretized	Number	ID
SL_AVN52	Signed Line of "AVN 52" Risk (valorized in case FLAG_PREMIUM_ _ALLOCATION = 'T')	Number	Float number from 0 to 1. (only positive value)
WL_AVN52	Writte Line of "AVN 52" Risk	Number	Float number from 0 to 1. (only positive value)
OF_AVN52	Offered Differential of "AVN 52" Risk	Number	Float number from 0 to 1.
FD_AVN52	Final Differential of "AVN 52" Risk	Number	Float number from 0 to 1.

Continued on next page

Table 7.4 – continued from previous page

Feature	Feature Description	Type	Value
DPA_AVN52	Deposit Premium Allocation of "AVN 52" Risk	Number	Float number from 0 to 1.
FLG_HWMAV	FLAG "Hull War Maximum Agreed Value" Risk presence	Number	1 : Presence of this risk type - 0 : Absence of this risk type
EXPOSURE_HWMAV	Number that represents the Maximum Agreed Value of the "Hull War Maximum Agreed Value" Risk	Number	Float number. (The value is expressed in the currency of the risk domicile: RISK_DOMICILE_ _COD_CURRENCY)
EXPOSURE_HWMAV_UNIFORM	Same as EXPOSURE_HWMAV %	Number	Float number. (The value is expressed in AED currency)
ID_BAND_EXPOSURE_ _HWMAV_UNIFORM	Same as EXPOSURE_HMWAV but the value is discretized	Number	ID
SL_HWMAV	Signed Line of "Hull War Maximum Agreed Value" Risk (valorized in case FLAG_PREMIUM_ _ALLOCATION = 'T')	Number	Float number from 0 to 1. (only positive value)
WL_HWMAV	Writte Line of "Hull War Maximum Agreed Value" Risk	Number	Float number from 0 to 1. (only positive value)
OF_HWMAV	Offered Differential of "Hull War Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
FD_HWMAV	Final Differential of "Hull War Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
DPA_HWMAV	Deposit Premium Allocation of "Hull War Maximum Agreed Value" Risk	Number	Float number from 0 to 1.
FLG_E_HULL	Flag Presence of Eastern Fleet - Hull Exposure	Number	1 : Presence - 0 : Absence
FLG_E_PPL	Flag Presence of Eastern Fleet - PPL Exposure	Number	1 : Presence - 0 : Absence
FLG_E_TPL	Flag Presence of Eastern Fleet - TPL Exposure	Number	1 : Presence - 0 : Absence
FLG_W_HULL	Flag Presence of Western Fleet - Hull Exposure	Number	1 : Presence - 0 : Absence
FLG_W_PPL	Flag Presence of Western Fleet - PPL Exposure	Number	1 : Presence - 0 : Absence
FLG_W_TPL	Flag Presence of Western Fleet - TPL Exposure	Number	1 : Presence - 0 : Absence
E_HULL	Exposure Level of Eastern Fleet - Hull Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
E_PPL	Exposure Level of Eastern Fleet - PPL Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)

Continued on next page

Table 7.4 – continued from previous page

Feature	Feature Description	Type	Value
E_TPL	Exposure Level of Eastern Fleet - TPL Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
W_HULL	Exposure Level of Western Fleet - Hull Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
W_PPL	Exposure Level of Western Fleet - PPL Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
W_TPL	Exposure Level of Western Fleet - TPL Exposure	Number	0 - 20 - 40 - 60 - 80 - 100 (see decode on table T_EXPOSURE _LEVEL)
ID_BUSINESS_STRUCTURE	Aggregate the features: "FLG_HMAV", "FLG_CSL", "FLG_PA", "FLG_AVN52", "FLG_HWMAV"	Number	ID (see decode on table T_SUP _BUSINESS_STRUCTURE)
ID_BUSINESS_RISK_EXPOSURE	Aggregate the features "FLG_E_HULL", "FLG_E_PPL", "FLG_E_TPL", "FLG_W_HULL", "FLG_W_PPL", "FLG_W_TPL", "E_HULL", "E_PPL", "E_TPL", "W_HULL", "W_PPL", "W_TPL"	Number	ID (see decode on table T_SUP_BUSINESS_EXPOSURE)

7.6.1 Subset of features

During the dataset analysis it is also define the follows different subset of features:

- **A - *subset_ UsefulFeatures*** : contains all features needed to reach our target.
- **B - *subset_ UsefulShortFeatures_ onlyOriginalFeatures*** : starting from subset_UsefulFeatures contains only the features which are extracted from the database (original features without feature engineering).
- **C - *subset_ UsefulShortFeatures_ withNewFeatures*** : contains all features of subset_UsefulShortFeatures_onlyOriginalFeatures plus the new features created.
- **D - *subset_ UsefulShortFeatures_ Compact*** : removes from subset_UsefulShortFeatures_withNewFeature all features whose information are aggregate in other features. This subset contains the minimum number of features completed by all information.

- **E - *subset_ Features_ Problem_ A*** : contains all significant features useful to solve the problema A. The goal of the business problem A is trying to help the user during a specific phase of the business. All features available and stable at that moment are reported on this subset.
- **F - *subset_ Features_ Problem_ B*** : contain all significant features useful to solve the problem B. The target of the business problem B is trying to help the user during a specific phase of the business. All the features available and stable at that moment are reported on this subset.
- **PA01 *subset_ UsefulFeatures_ Problem_ A*** : contains all features which are both on subset *subset_ UsefulFeatures* and subset *subset_ Features_ Problem_ A*.
- **PA02 *subset_ UsefulShortFeatures_ onlyOriginalFeatures_ Problem_ A*** : contains all features which are both on subset *subset_ UsefulShortFeatures_ onlyOriginalFeatures* and subset *subset_ Features_ Problem_ A*.
- **PA03 *subset_ UsefulShortFeatures_ withNewFeatures_ Problem_ A*** : contains all features which are both on subset *subset_ UsefulShortFeatures_ withNewFeatures* and subset *subset_ Features_ Problem_ A*.
- **PA04 *subset_ UsefulShortFeatures_ Compact_ Problem_ A*** : contains all features which are both on subset *subset_ UsefulShortFeatures_ Compact* and subset *subset_ Features_ Problem_ A*.
- **PB01 *subset_ UsefulFeatures_ Problem_ B*** : contains all features which are both on subset *subset_ UsefulFeatures* and subset *subset_ Features_ Problem_ B*.
- **PB02 *subset_ UsefulShortFeatures_ onlyOriginalFeatures_ Problem_ B*** : contains all features which are on the subset both *subset_ UsefulShortFeatures_ onlyOriginalFeatures* and subset *subset_ Features_ Problem_ B*.
- **PB03 *subset_ UsefulShortFeatures_ withNewFeatures_ Problem_ B*** : contains all features which are on thesubset both *subset_ UsefulShortFeatures_ withNewFeatures* and subset *subset_ Features_ Problem_ B*.
- **PB04 *subset_ UsefulShortFeatures_ Compact_ Problem_ B*** : contains all the features which are on the subset both *subset_ UsefulShortFeatures_ Compact* and subset *subset_ Features_ Problem_ B*.

Table 7.5: List of different features subsets identified

Feature	A	B	C	D	E	F
ID_BUSINESS	✓	✓	✓	✓		
COD_PHASE						
BUSINESS_STATUS						
ID_CLASS	✓	✓	✓	✓	✓	✓
CLASS					✓	✓
IS_ID_CLASS_1	✓		✓		✓	✓
IS_ID_CLASS_2	✓		✓		✓	✓
IS_ID_CLASS_3	✓		✓		✓	✓
IS_ID_CLASS_4	✓		✓		✓	✓
IS_ID_CLASS_5	✓		✓		✓	✓
IS_ID_CLASS_6	✓		✓		✓	✓
IS_ID_CLASS_7	✓		✓		✓	✓
IS_ID_CLASS_8	✓		✓		✓	✓
IS_ID_CLASS_9	✓		✓		✓	✓
IS_ID_CLASS_10	✓		✓		✓	✓
IS_ID_CLASS_11	✓		✓		✓	✓
IS_ID_CLASS_12	✓		✓		✓	✓
IS_ID_CLASS_13	✓		✓		✓	✓
IS_ID_CLASS_14	✓		✓		✓	✓
IS_ID_CLASS_15	✓		✓		✓	✓
IS_ID_CLASS_16	✓		✓		✓	✓
IS_ID_CLASS_17	✓		✓		✓	✓
IS_ID_CLASS_19	✓		✓		✓	✓
IS_ID_CLASS_19	✓		✓		✓	✓
IS_ID_CLASS_20	✓		✓		✓	✓
ID_SOURCE	✓	✓	✓	✓	✓	✓
SOURCE					✓	✓
IS_ID_SOURCE_1	✓		✓		✓	✓
IS_ID_SOURCE_2	✓		✓		✓	✓
IS_ID_SOURCE_3	✓		✓		✓	✓
IS_ID_SOURCE_4	✓		✓		✓	✓
IS_ID_SOURCE_5	✓		✓		✓	✓
IS_ID_SOURCE_6	✓		✓		✓	✓
STATUS						✓
IS_STATUS_RENEWAL	✓	✓	✓	✓		✓
IS_STATUS_NEW	✓		✓			✓
PREVIOUS_POLICY_NUMBER						✓

Continued on next page

Table 7.5 – continued from previous page

Feature	A	B	C	D	E	F
FIRST_ID_BUSINESS	✓	✓	✓	✓		✓
UNDERWRITING_YEAR	✓	✓	✓	✓		
MENA_NON_MENA						
IS_MENA	✓	✓	✓	✓		
IS_NON_MENA	✓		✓			
BUSINESS_COD_CURRENCY					✓	✓
BUSINESS_EXCHANGE_RATE					✓	✓
INCEPTION	✓	✓				
EXPIRY	✓	✓				
N_GG	✓		✓	✓		
GROSS_PREMIUM_100	✓	✓	✓		✓	✓
GROSS_PREMIUM_100_UNIFORM	✓	✓	✓		✓	✓
ID_BAND_GROSS_PREMIUM_100_UNIFORM	✓		✓	✓	✓	✓
POLICY_NUMBER						
FLAG_PREMIUM_ALLOCATION	✓	✓	✓	✓		
D_P_SIGNED_LINE	✓	✓	✓	✓		
ID_PROVIDER	✓	✓	✓	✓	✓	✓
PROVIDE					✓	✓
ID_CLIENT	✓	✓	✓	✓	✓	✓
CLIENT					✓	✓
ID_COUNTRY	✓	✓	✓	✓	✓	✓
COUNTRY					✓	✓
COUNTRY_REGION					✓	✓
IS_COUNTRY_REGION_01	✓		✓		✓	✓
IS_COUNTRY_REGION_02	✓		✓		✓	✓
IS_COUNTRY_REGION_03	✓		✓		✓	✓
IS_COUNTRY_REGION_04	✓		✓		✓	✓
IS_COUNTRY_REGION_05	✓		✓		✓	✓
IS_COUNTRY_REGION_06	✓		✓		✓	✓
IS_COUNTRY_REGION_07	✓		✓		✓	✓
IS_COUNTRY_REGION_08	✓		✓		✓	✓
IS_COUNTRY_REGION_09	✓		✓		✓	✓
IS_COUNTRY_REGION_10	✓		✓		✓	✓
IS_COUNTRY_REGION_11	✓		✓		✓	✓
IS_COUNTRY_REGION_12	✓		✓		✓	✓
IS_COUNTRY_REGION_13	✓		✓		✓	✓
RISK_DOMICILE_COD_CURRENCY					✓	✓

Continued on next page

Table 7.5 – continued from previous page

Feature	A	B	C	D	E	F
RISK_DOMICILE_EXCHANGE_RATE					✓	✓
ID_BUSINESS_STRUCTURE	✓			✓	✓	✓
FLG_HMAV	✓		✓	✓	✓	✓
EXPOSURE_HMAV	✓	✓	✓		✓	✓
EXPOSURE_HMAV_UNIFORM	✓	✓	✓		✓	✓
SL_HMAV	✓	✓	✓	✓		
WL_HMAV	✓	✓	✓	✓		
OF_HMAV	✓	✓	✓	✓	✓	✓
FD_HMAV	✓	✓	✓	✓		
DPA_HMAV	✓	✓	✓	✓		
ID_BAND_EXPOSURE_HMAV_UNIFORM	✓		✓	✓	✓	✓
FLG_CSL	✓		✓	✓	✓	✓
EXPOSURE_CSL	✓	✓	✓		✓	✓
EXPOSURE_CSL_UNIFORM	✓	✓	✓		✓	✓
SL_CSL	✓	✓	✓	✓		
WL_CSL	✓	✓	✓	✓		
OF_CSL	✓	✓	✓	✓	✓	✓
FD_CSL	✓	✓	✓	✓		
DPA_CSL	✓	✓	✓	✓		
ID_BAND_EXPOSURE_CSL_UNIFORM	✓		✓	✓	✓	✓
FLG_PA	✓		✓	✓	✓	✓
EXPOSURE_PA	✓	✓	✓		✓	✓
EXPOSURE_PA_UNIFORM	✓	✓	✓		✓	✓
SL_PA	✓	✓	✓	✓		
WL_PA	✓	✓	✓	✓		
OF_PA	✓	✓	✓	✓	✓	✓
FD_PA	✓	✓	✓	✓		
DPA_PA	✓	✓	✓	✓		
ID_BAND_EXPOSURE_PA_UNIFORM	✓		✓	✓	✓	✓
FLG_AVN52	✓		✓	✓	✓	✓
EXPOSURE_AVN52	✓	✓	✓		✓	✓
EXPOSURE_AVN52_UNIFORM	✓	✓	✓		✓	✓
SL_AVN52	✓	✓	✓	✓		
WL_AVN52	✓	✓	✓	✓		
OF_AVN52	✓	✓	✓	✓	✓	✓
FD_AVN52	✓	✓	✓	✓		
DPA_AVN52	✓	✓	✓	✓		

Continued on next page

Table 7.5 – continued from previous page

Feature	A	B	C	D	E	F
ID_BAND_EXPOSURE_AVN52_UNIFORM	✓		✓	✓	✓	✓
FLG_HWMAV	✓		✓	✓		
EXPOSURE_HWMAV	✓	✓	✓			
EXPOSURE_HWMAV_UNIFORM	✓	✓	✓			
SL_HWMAV	✓	✓	✓	✓		
WL_HWMAV	✓	✓	✓	✓		
OF_HWMAV	✓	✓	✓	✓		
FD_HWMAV	✓	✓	✓	✓		
DPA_HWMAV	✓	✓	✓	✓		
ID_BAND_EXPOSURE_HWMAV_UNIFORM	✓		✓	✓		
ID_BUSINESS_RISK_EXPOSURE	✓		✓	✓	✓	✓
FLG_E_HULL	✓		✓	✓	✓	✓
FLG_E_PPL	✓		✓	✓	✓	✓
FLG_E_TPL	✓		✓	✓	✓	✓
FLG_W_HULL	✓		✓	✓	✓	✓
FLG_W_PPL	✓		✓	✓	✓	✓
FLG_W_TPL	✓		✓	✓	✓	✓
E_HULL	✓	✓	✓	✓	✓	✓
E_PPL	✓	✓	✓	✓	✓	✓
E_TPL	✓	✓	✓	✓	✓	✓
W_HULL	✓	✓	✓	✓	✓	✓
W_PPL	✓	✓	✓	✓	✓	✓
W_TPL	✓	✓	✓	✓	✓	✓

7.7 Analysis of the Dataset

The plots reppresented in Figure 7.19 have been made after the Data cleaning. The Feature engineering phases have to be compared with the plots rappresented in Figure 7.3.

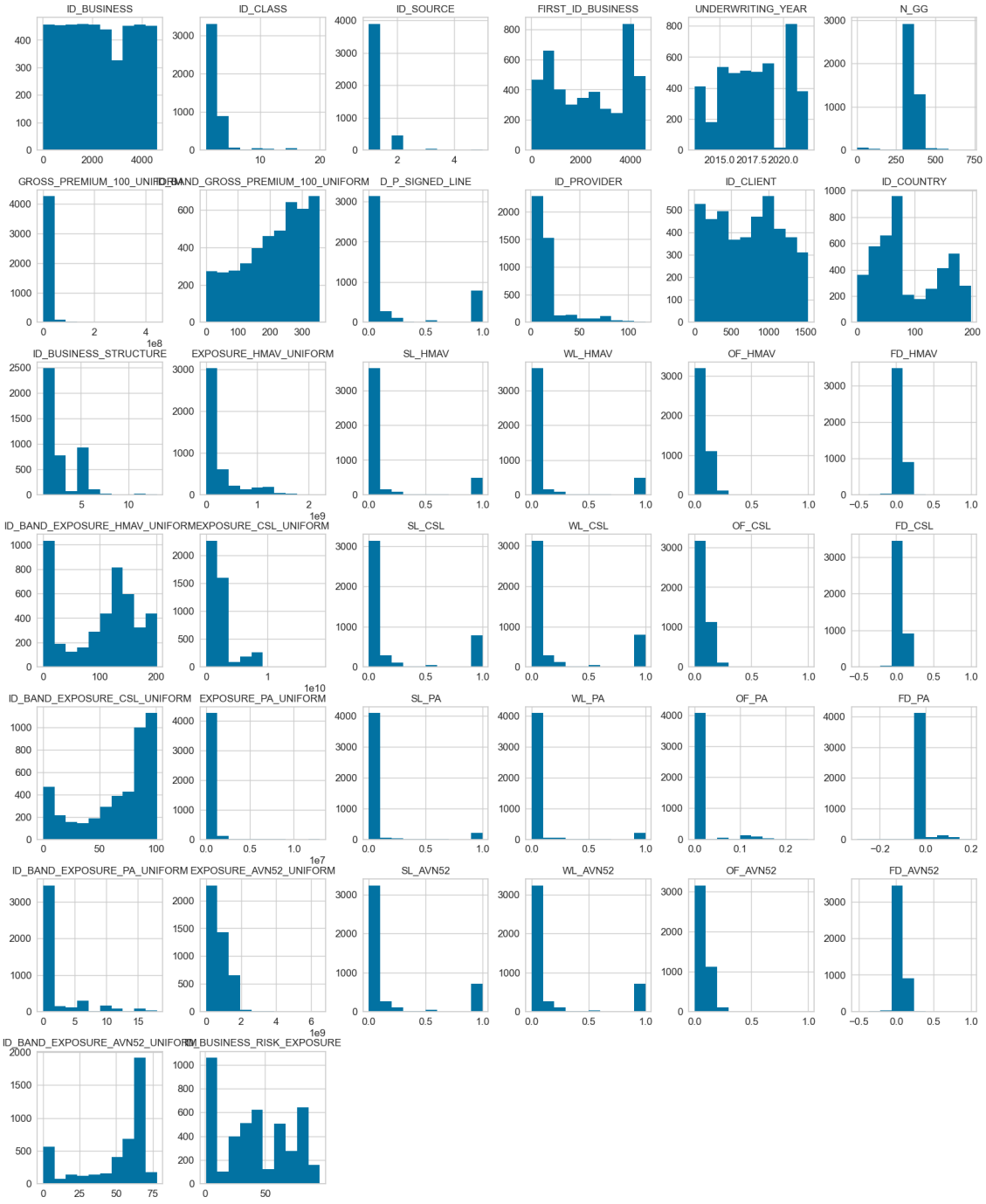

Figure 7.19: Features distribution

As an example, in the Figure 7.20 the "GROSS_PREMIUM_100" feature extracted from the datatbase is compared with the new features "ID_BAND_GROSS_PREMIUM_100_UNIFORM" after the Data cleaning and Feature engineering phases.

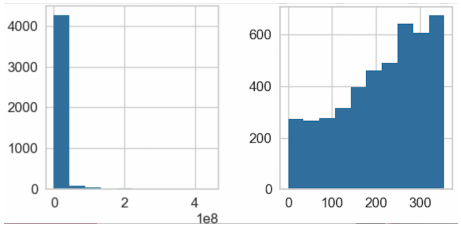

Figure 7.20: Compares "GROSS_PREMIUM_100" feature distribution

The plots rappresented in the Figure 7.21 which have been made after the Data cleaning and Feature engineering phases have to be compared with the plots rappresented in the Figure 7.2.

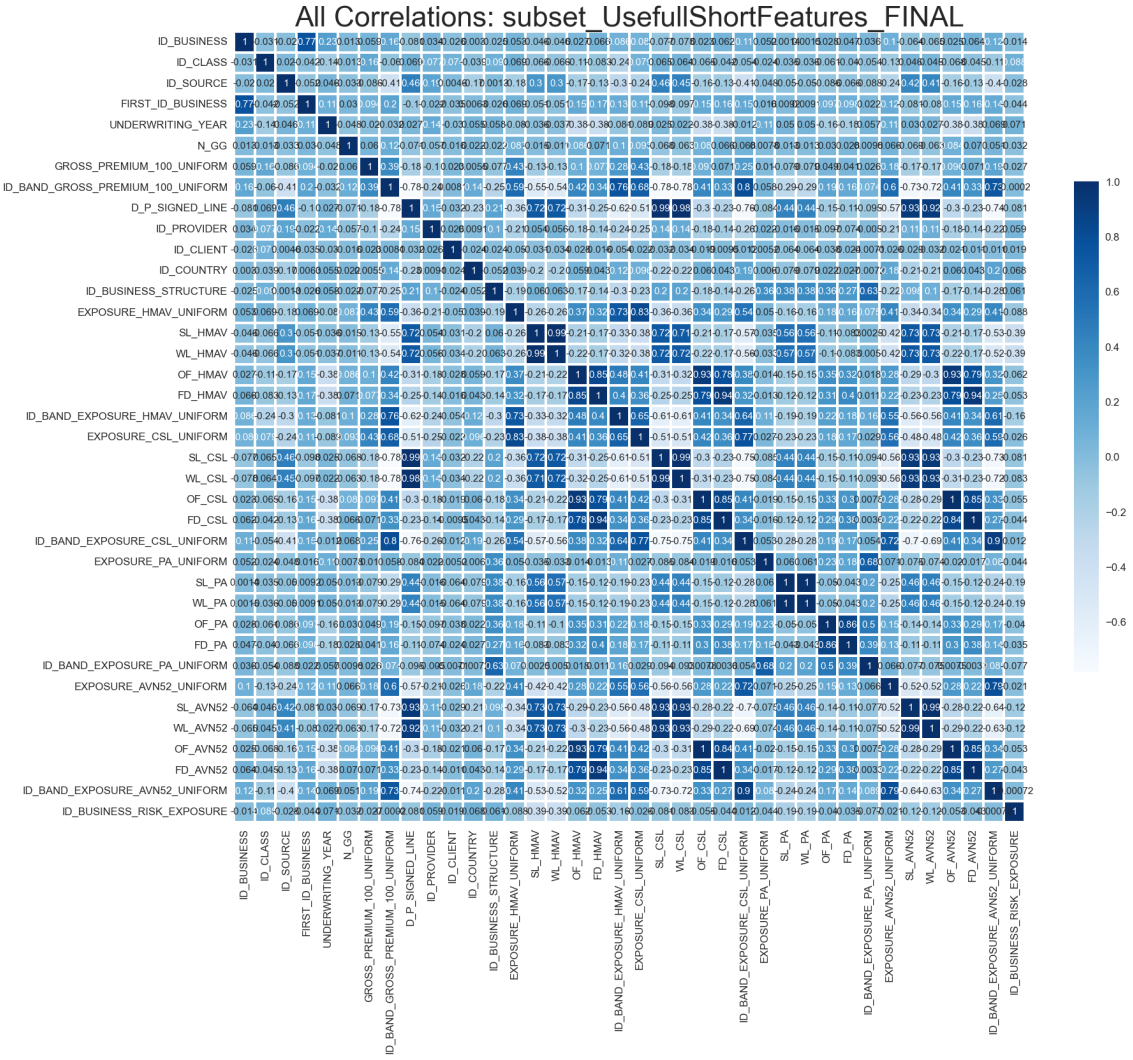

Figure 7.21: Cross correlations

7.8 Silhouette analysis

A silhouette analysis helps us to provide us with an idea of compactness and good separation in one mathematical equation. This equation calculates how well a particular data example fits within a cluster as compared to its neighboring clusters. Each example assigned a value, called a silhouette coefficient or silhouette score, between -1 and 1.
This value indicates the following:

- A high coefficient means that the example is far away from neighboring clusters; therefore fits well within its own cluster.
- A coefficient close to 0 means that the example is near the decision boundary between clusters.
- A coefficient in the negative means that the example is closer to a neighboring cluster than its own; therefore it is likely it has been placed in the wrong cluster.

The ideal is to have our data examples as close as possible to 1. Low coefficients indicate that your k value (number of clusters) needs to be adjusted.

The python function named KElbowVisualizer from yellowbrick.cluster package allow us to :

- calculate the coefficient for each data example; then group each data into its respective cluster;
- calculate the average coefficient of each cluster;
- calculate the average of the entire model given k (number of clusters);
- plotted on a graph the values derived from a silhouette analysis;

Looking at the graph plotted: for each cluster, the data example with the highest coefficient is on top, with the lowest coefficient at the bottom. This forms a silhouette-like shape for each cluster. The cluster with the highest average coefficient is often placed on top, and the rest of clusters are placed in descending order.

7.9 Model to solve the Business Problem A

As already said in the chapter 6, as to solve the business problem A a Recommender System realized by using a clustering unsupervised machine learning algorithm is needed.
The python library called scikit-learn[11] let compare different types of algorithms that are reported on Table 6.1, easily.
Because:

- an Inductive machine learning is needed[1]
- our data refers to flat geometry [2]

the useful algorithms reported on Table 6.1 are the following:

- **Gaussian mixtures** The Gaussian Mixture object implements the expectation-maximization (EM) algorithm for fitting mixture-of-Gaussian models.
- **k-Means** The k-Means algorithm clusters data by separating samples in n groups of equal variance, minimizing with a criterion known as the inertia or within-cluster sum-of-squares. This algorithm requires that the number of clusters is specified.
- **Bisecting k-Means** The Bisecting k-Means is an iterative variant of k-Means.

Considering the high number of clusters expected, another algorithm taken into consideration is the Affinity Propagation algorithm.

- **Affinity Propagation** creates clusters by sending messages between pairs of samples until convergence. Although the usecase of this algorithm is non-flat geometry, it will be tested since works well with a high number of clusters, which is our case.

[1] This equals to build a model with some data which can be applicable to new instances.
[2] Non-flat geometry clustering is useful when the clusters have a specific shape, i.e. a non-flat manifold, and the standard euclidean distance is not the right metric.

7.9.1 Gaussian mixtures algorithm

Gaussian mixtures is the first algorithm tested, but unfortunately it proved not to be useful in our case due to our dataset.
On Figure 7.22 is rappresented the silhouette plot of the best model obtained with 3 clusters and a silhouette score equals to -0.016538102986912266.

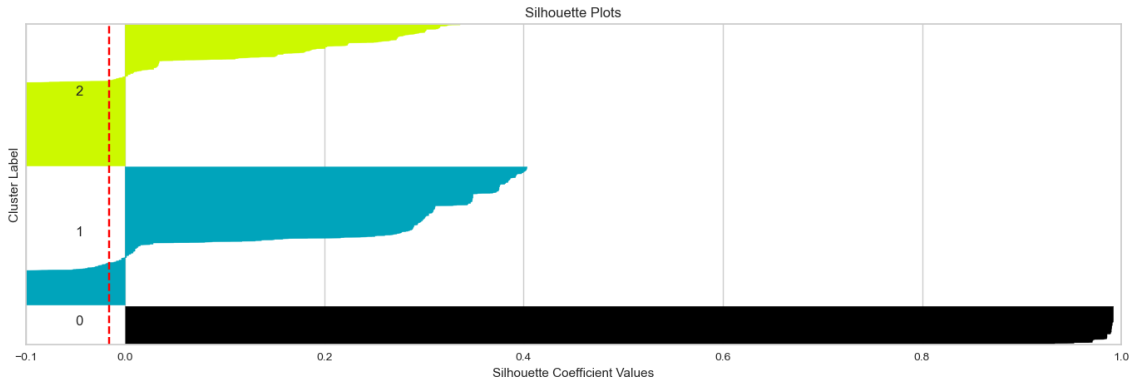

Figure 7.22: Silhouette plot: Gaussian mixtures algorithm - Best case

During this work thesis different combination of input parameter for the algorithm used have been tested; the parameters reported are the best combination found.

```
model = GaussianMixture(
n_components=cluster_count,
random_state=0, reg_covar=1e-05,
init_params='random_from_data')
```

The algorithm tunning phase was not efficient to increase the number of clusters or even improve the silhouette score; whilst the k-Means algorithm was capable to perform better without any parameters tunning straigh from the beginning.
Due to this factors, the work thesis is focused on k-Means and Bisecting k-Means algorithms.

7.9.2 k-Means and Bisecting k-Means algorithms

k-Means algorithm

k-Means is one of the most popular clustering algorithm and is applicable to many common types of clustering problems and datasets. It is and algorithm for unsupervised machine learning that groups like data examples together for the purpose of revealing patterns in the data.
It makes this possible by defining a set of k groups (clusters). Each data example is placed within the cluster whose center (called a centroid) is the closest to that data example. Closeness can be defined by a distance metric that is chosen during training.
Once the algorithm assigns data examples to clusters, it recomputes each centroid by calculating the mean of all data examples in the centroid's cluster. This process continues until the data examples no longer change clusters (i.e., the k-means converge), or until a specific number of iteration has been met.
The k-means clustering algorithm should not be confused with the k-nearest neighbor algorithm.[3]

The ultimate goal of k-means clustering is the minimization of costs, as most of other machine learning algorithms. The global cost function $J(\theta)$ can be written as follows:

$$J(\theta) = \sum_{i=1}^{n} min_j(dist(x_i, c_j))$$

where

- n is the total number of examples
- x_i is the i^{th} data example
- c_j is the j^{th} centroid

So, from right to left, the distance between a data example and a centroid is taken (dist). Then, the centroid with the minimum (nearest) distance to the example is taken (min). Lastly, the sum of all nearest distances is calculated. Ultimately, this helps to find the centroids that minimize this total distance. However, it is not feasible applying this optimization usually.

As an example, with an n of 25 and 4 as the number of clusters, there are roughly 47 trillion possible assignments.
This is why k-means clustering is an iteractive algorithm that requires local optimization. The process follows this pattern:

[3] k-means clustering algorithm is used to solve unsupervised problems by clustering data into groups whilst k-nearest neighbor algorithm is used to solve supervised problems.

1. It starts by taking the number of desired clusters, then it randomly assigns centroids for each cluster.

2. therefore, it assigns each data example to whatever centroid is currently closer. This is the same as minimizing the cost of these assignments.

3. Furthermore, the algorithm moves each centroid so that it is in the center of the data samples that were assigned to it. This is effectively the same as minimizing the cost of the centroids.

4. The process is repeated until convergence or until an iteration maximum is met.

As a consequence, this iterative cost minimizing scheme is a more efficient approach to optimization. However, it doesn not always achieve the perfect global optimization, especially if the initial randomly selected centroids were placed in sub-optimal locations. Hence, the algorithm implementation on python let to re-initialize the k-means algorithm with different randomly chosen centroids to overcome this.

Bisecting k-Means algorithms

The Bisecting k-Means is an iteractive variant of k-Means, using divisive hierarchical clustering.
Instead of creating all centroids at once, centroids are picked up progressively on the base of a previous clustering: a cluster is split into two new clusters repeatedly until the target number of clusters is reached.

Number of clusters

These two algorithms requires the number of clusters to be specified as an input. Because of that, the possible approach are:

- uses the Elbow method to determine the number of cluster
- does the same calculations for different k values and compare the results of the different k values

Elbow method

In k-means clustering, the primary challenge is determining k (number of clusters). There are several evaluation metrics which help to determine the optimal k as the elbow method presented in this section.

One method of determining k (the optimal number of cluster) is to calculate the mean distance bewtweeen each data example and its associated centroid. As k increases, the mean distance decreases necessarily. However, at some point, increasing

k any further becomes pointless and does not reduce the mean distance in any significant way. The point at which the mean distance no longer decreases in a significant way is called the *elbow point*, and it is usually a good indicator of what k should be.

On Figure 7.23 is reported the Elbow chart to determine the optimal number of clusters with the features subset "PA01 subset_UsefulFeatures_Problem_A" and the follows parameters:

```
KElbowVisualizer(
KMeans(init='k-means++', random_state = 10, n_init=10),
k=(1, 100) )
```

As visible on Figure 7.23 the elbow is found at k=6 clusters with a Distortion Score which equals to 1373171940042768121856,000.

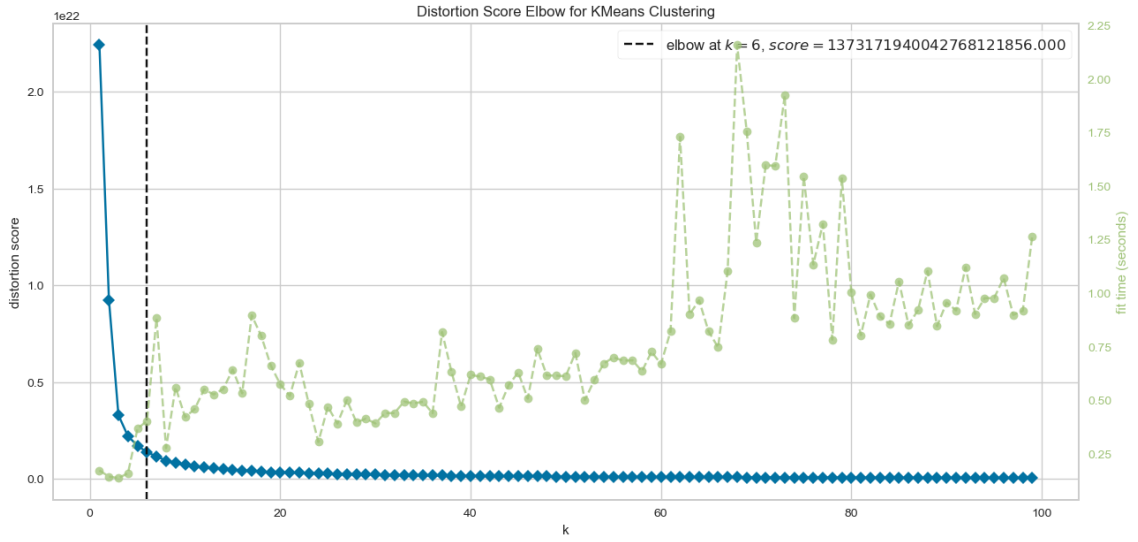

Figure 7.23: Elbow method to determine the optimal number of clusters (subset: PA01 subset_UsefulFeatures_Problem_A)

On Figure 7.24 is reported the Elbow chart to determine the optimal number of clusters with the features subset "PA01 subset_UsefulFeatures_Problem_A" and the follows parameters:

```
visualizer = KElbowVisualizer(
KMeans(init='k-means++', random_state = 10, n_init=10),
k=(1, 2500) )
```

As visible on Figure 7.24 the elbow is found at k=26 clusters with a Distortion Score which equals to 2500714471757411450 88,000.

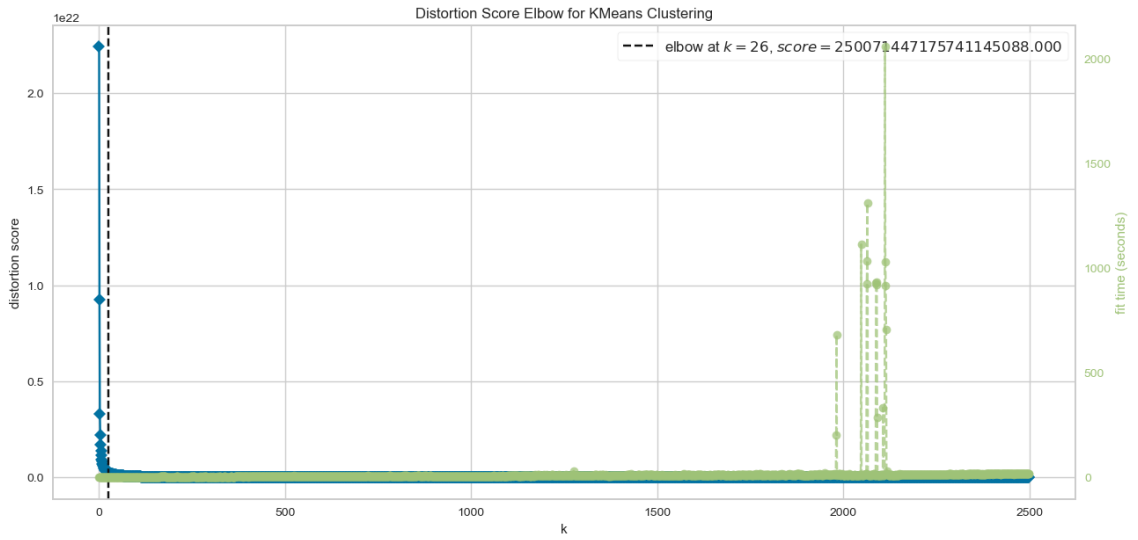

Figure 7.24: Elbow method to determine the optimal number of clusters (subset: PA01 subset_UsefulFeatures_Problem_A)

The two plots reported on Figures 7.23 and 7.24 show two different elbow points, but the parameters of the k-Means algorithm are the same and the only parameter change is the dimension of the band related to the number of cluster to be tested. This means that the "elbow method" can be useful as a starting point during the analysis, but it cannot provide us with a precise absolute number of clusters.

Silhouette score analysis

Another approach to estimate the correct number of cluster it is to compute the silhouette score for a wide range of possible number of clusters (in our case from 2 to 2500 cluster) as to analyze the results.
The features subsets[4] used to compute the silhouette score at different values of k (number of clusters given as input to the algorithm) are:

- PA01 : subset_Problem_A_UsefulFeatures
- PA02 : subset_Problem_A_UsefulShortFeatures_onlyOriginalFeatures
- PA03 : subset_Problem_A_UsefulShortFeatures_withNewFeature
- PA04 : subset_Problem_A_UsefulShortFeatures_Compact

During this work thesis different combination of input parameter for the algorithms used have been tested. The parameters reported here with are the best combinations we found.

[4]See the chapter 7.6.1

On Figure 7.25 are rappresented the values of the Silhouette scores computed for differents number of clusters with a k-Means algorithm runned with the following parameters:

```
model=KMeans(n_clusters=k, init='k-means++', random_state=42, n_init=100)
```

On Figure 7.26 are rappresented the values of the Silhouette scores computed for differents number of clusters with a Bisectingk-Means algorithm runned with the following parameters:

```
model=BisectingKMeans(n_clusters=k, random_state=42, n_init=100)
```

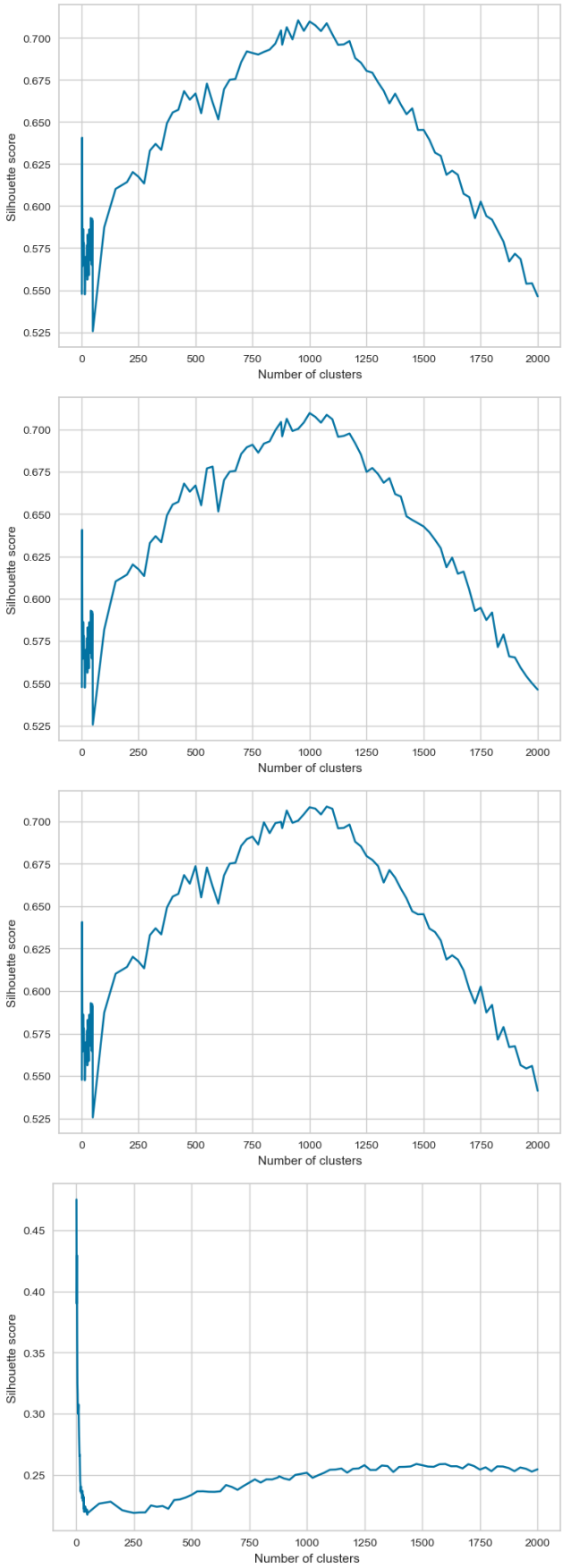

Figure 7.25: Silhouette analysis with k-Means algorithm (Subset: PA01, PA02, PA03, PA04).

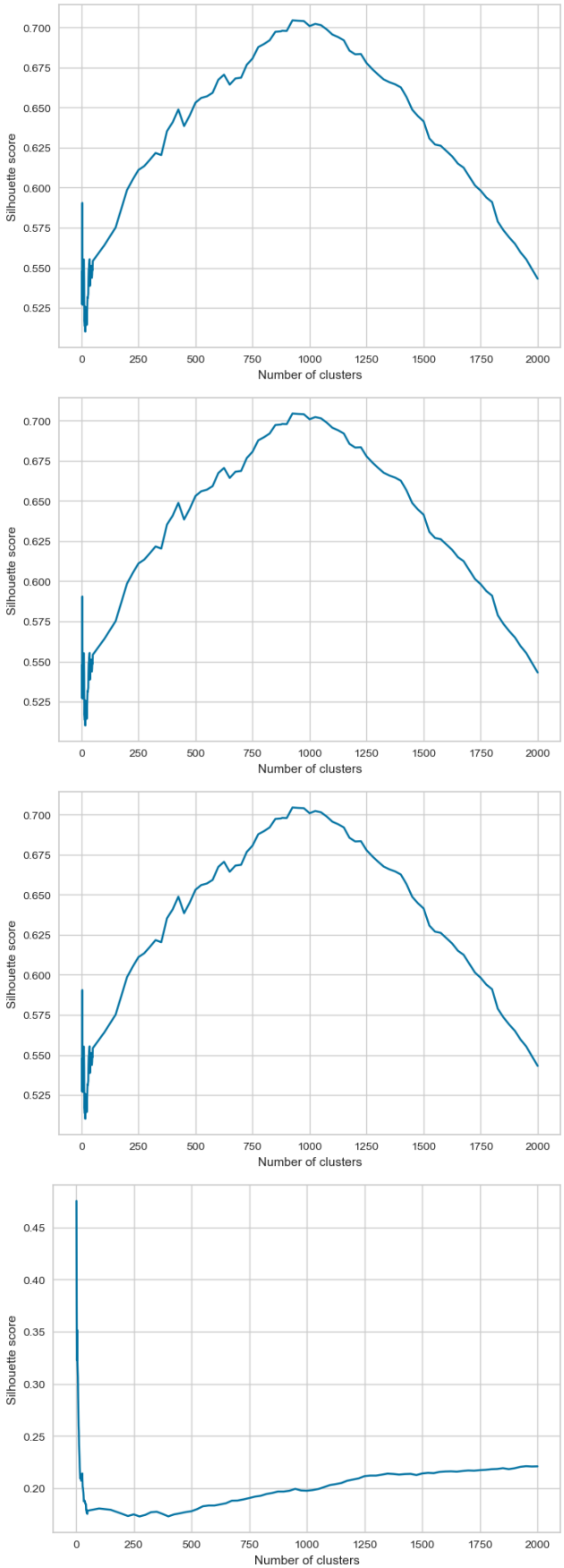

Figure 7.26: Silhouette analysis with Bisecting k-Means algorithm (Subset: PA01, PA02, PA03, PA04).

The chart on Figure 7.27 reports all data (it comes from the Figure 7.25 and the Figure 7.26) added together as to allow an easy comparison whilst the charts on Figures 7.28 and 7.29 report a focus on data observation from 0 to 100 clusters.

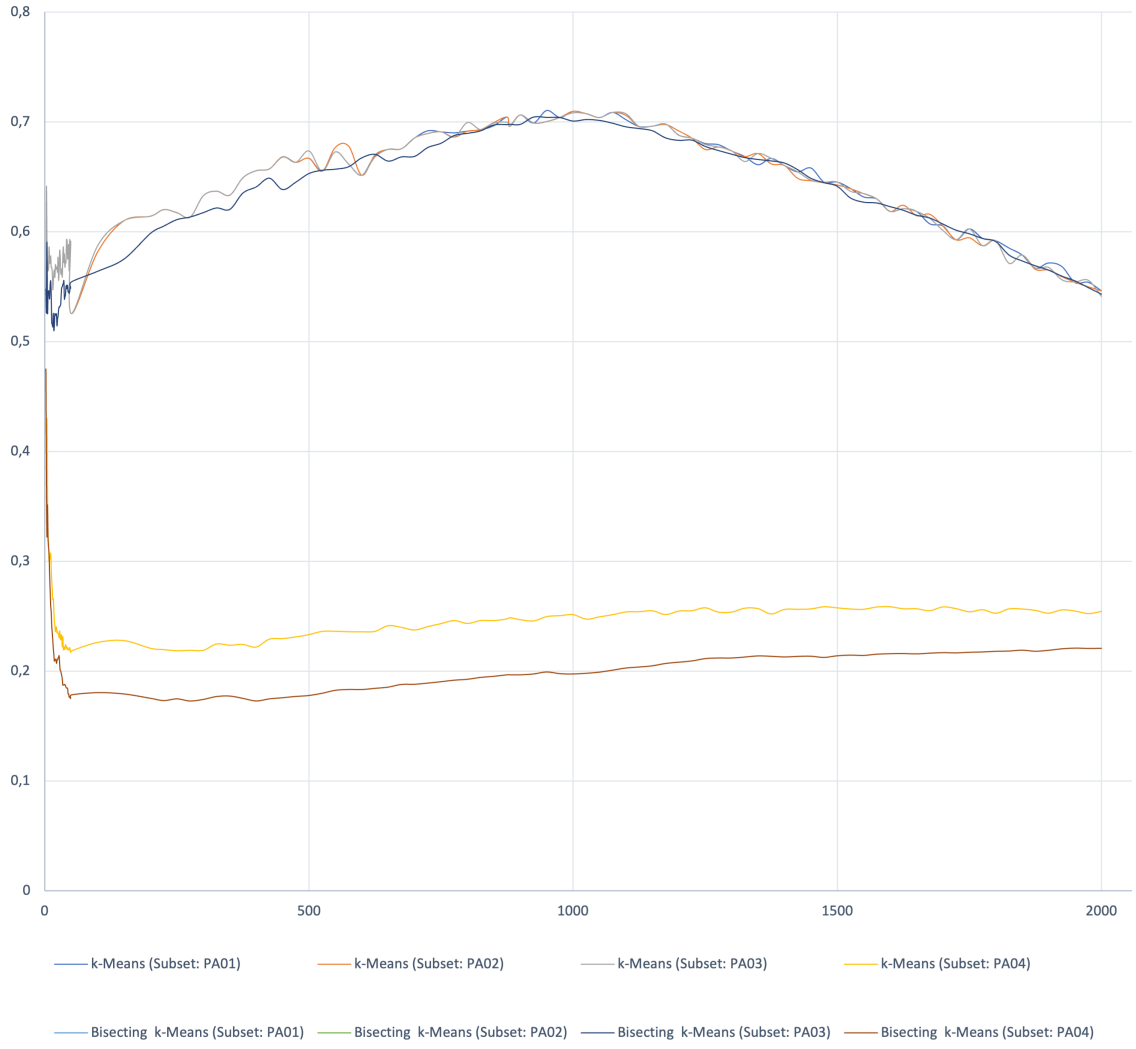

Figure 7.27: Summary of silhouette score analysis

Starting from the chart on Figure 7.27 the observations are the following:

- the feature subset "PA04 : subset_Problem_A_UsefulShortFeatures_Compact" is not a good feature subset, hence it can be removed from the future analysis;
- the shapes of silhouette score related to the features subsets utilized(PA01, PA02, PA03) are very similar (general trand);
- it is possible to identify 2 maximum values: the first is closed to 5/10 clusters (local maximum) whilst the second is closed to 800/1000 clusters (global maximum);

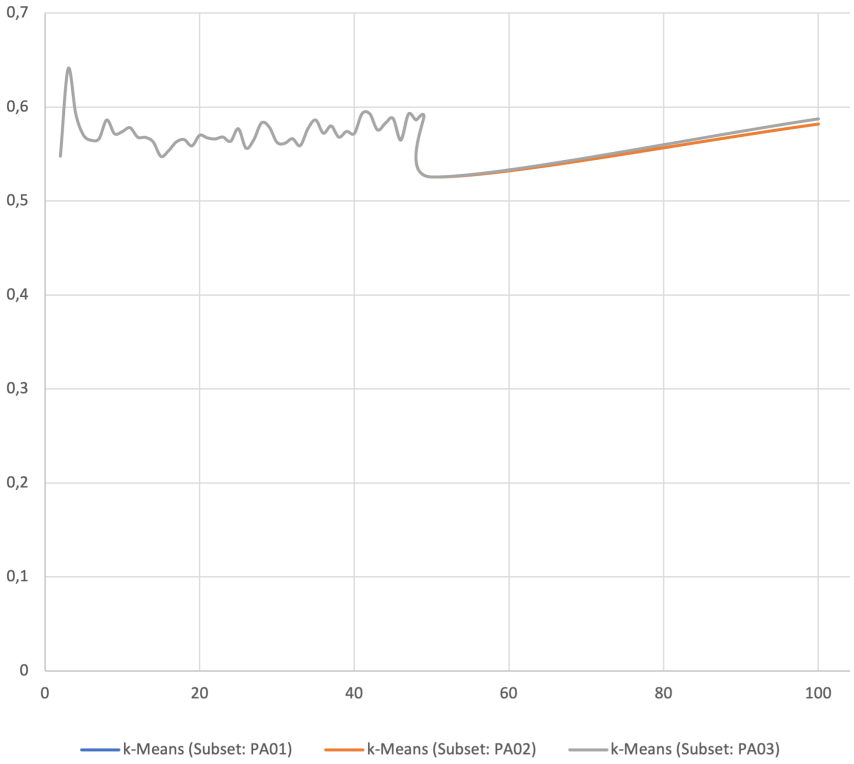

Figure 7.28: Focus on silhouette score analysis with k-Means algorithm

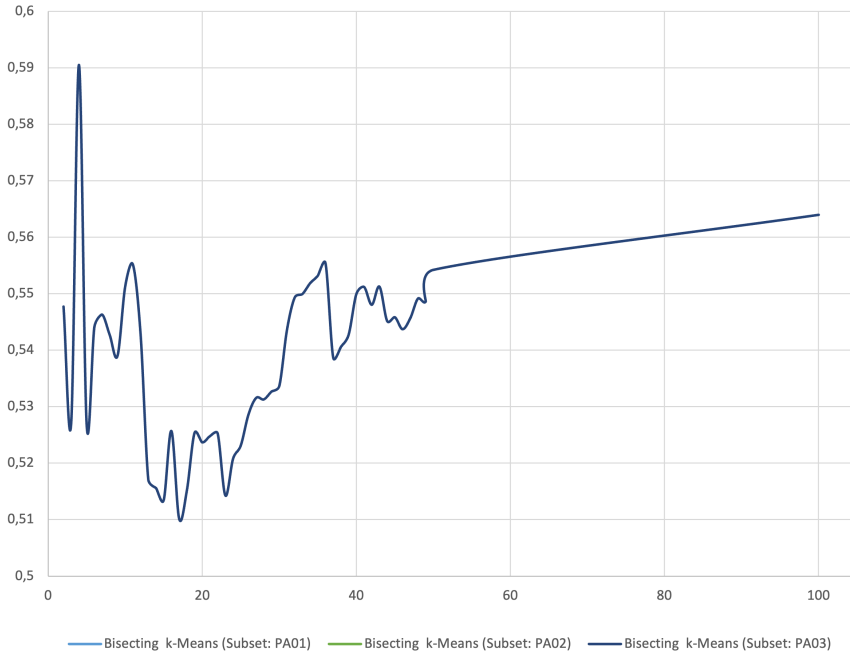

Figure 7.29: Focus on silhouette score analysis with Bisecting k-Means algorithm

Starting from the charts on Figures 7.25 and 7.26, the observations are the following:

- the shapes of silhouette score related to the features subsets utilized are very similars (general trand) for each and every type of algorithm;

- it is not possible to identify an exact local maximum;
- it is possible to consider with a certain approximation as local maximum for k-Means algorithm a number of cluster which equals to 8 with a silhouette score which equals to 0,58619821;
- it possible to consider with approximation as local maximun for Bisecting k-Means algorithm a number of cluster which equals to 4 with a silhouette score which equals to 0,590508591;

Low number of cluster

Low number of cluster means that the "Recommender System" realized should return a high number of passed business, whilst the expectation of the Business unit it is focuse on a system capable to return a group of maximum 10/20 past business for each and every business phase.
As consequence of the above considerations:

- the numbers of local clusters identified (8 and 4 clusters) on Figures 7.25 and 7.26 are not useful to reach our goal once related to the Business Problem A;
- the input of the number of clusters given to the final model will be picked up from the global maximum identified;

Is overfitting a problem in a unsupervised learning?

Analyzing the chart on Figure 7.27 and selecting the global maximum value (around 800/1000 clusters) a doubt may arise: is "the local maximum the right value of k (number of cluster) or the global maximum leads to a model which overfits?"

Overfitting means that our algorithm finds attributes patterns which exists in this dataset only; whilst it does not generalize to new and/or unseen data. In addition to find some real patterns, when overfitting, the algorithm finds also "patterns" that are mere stochastic noise. We define overfitting when some models perform better on training samples rather then on validation samples.
First of all, how can overfitting be defined for unsupervised learning? If a clustering analysis of data is perfomed, then there is no objective criteria to say that some outputs are "correct". In addition, there is no "correct" clustering solution, as there are no labels in a unsupervised scenario.
How can a clustering performance be evaluated? How can we say that it performs "worse" on a validation sample? The same applies to cross-validation.
On the other hand, it can be defined as overfitting within a unsupervised case such as a case where n clusters is fitted to n cases.
The same consideration can be made with clusters density estimation. There is

not a single "correct" solution. On the other hand, if it is set bandwidth in kernel density estimation to zero, you will end up with density estimate that fits perfectly to your data, but it does not translate to external data.

The trick here it is to find a solution that it is general enough to be useful, and detailed enough to share some specific features of our data ,but there is no single best solution like this.

Silhouette Plot

During the making model's phase it has been tested a "visual analyses", also. In order to perform such an activity, two new features have been defined TYPE_BUSINESS, RISK_BUSINESS which have been used to rappresent the values clusterized on a 2 dimensional charts. During this phase a lot of silhouette plots and data clusterized on a 2 dimensional chart have been analyzed. Some of those have been reported on the followiong figures: 7.30, 7.31, 7.32, 7.33, 7.34, 7.35, 7.36.

TYPE_BUSINESS = IS_STATUS_RENEWAL*50 + ID_CLIENT + ID_SOURCE + ID_CLASS*5 + ID_PROVIDER*10

RISK_BUSINESS = 5*(ID_COUNTRY + FLG_HMAV*5 + FLG_CSL*10 + FLG_PA*15 + FLG_AVN52*20)

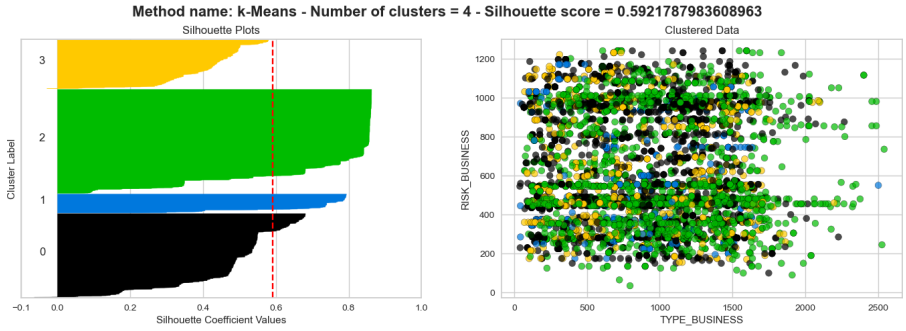

Figure 7.30: Silhouette Plot with k-Means algorithm - Feature subset: PA01 - K : 4

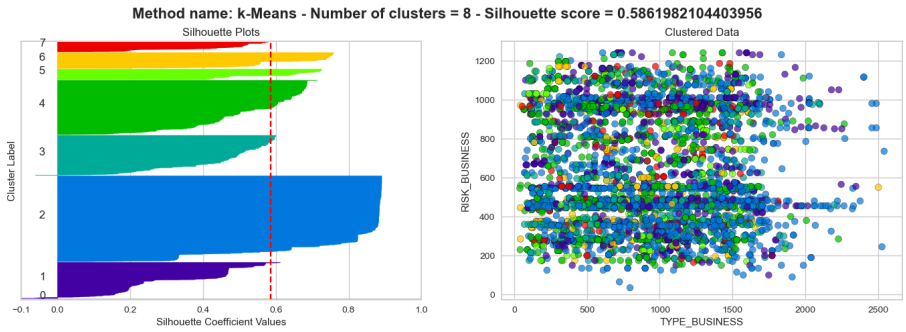

Figure 7.31: Silhouette Plot with k-Means algorithm - Feature subset: PA01 - K : 8

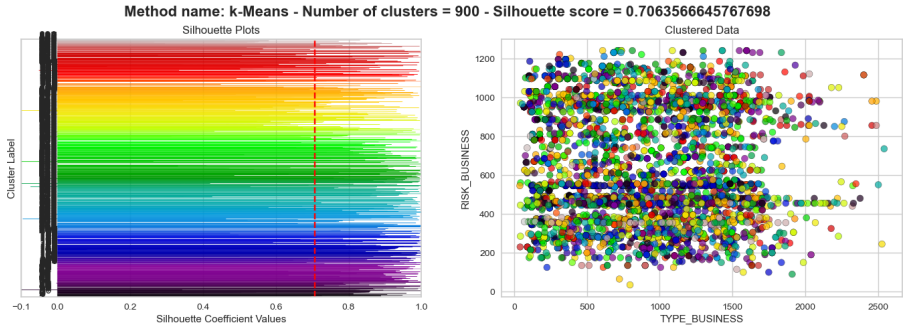

Figure 7.32: Silhouette Plot with k-Means algorithm - Feature subset: PA01 - K : 900

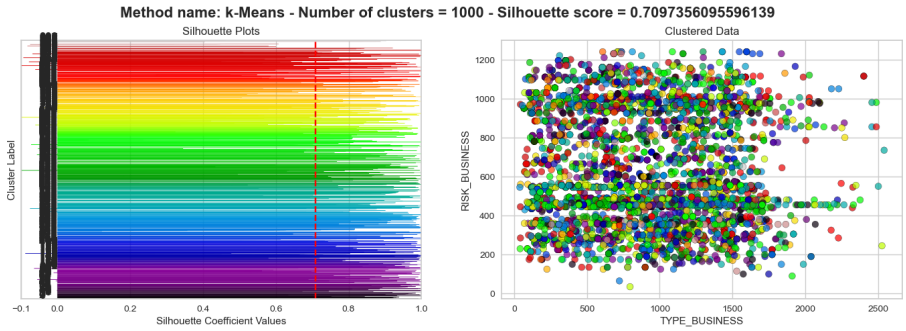

Figure 7.33: Silhouette Plot with k-Means algorithm - Feature subset: PA02 - K : 1000

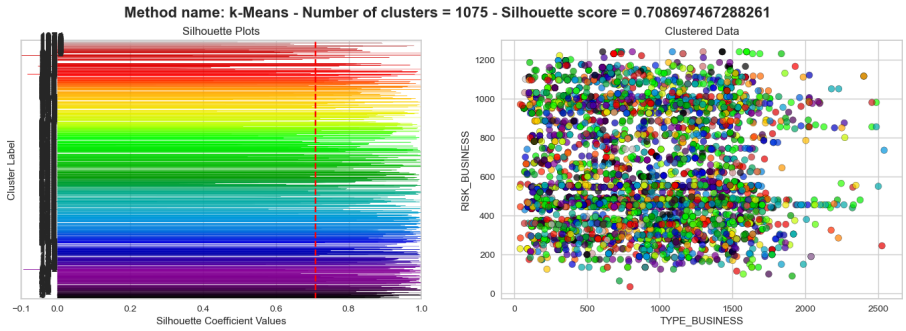

Figure 7.34: Silhouette Plot with k-Means algorithm - Feature subset: PA03 - K : 1075

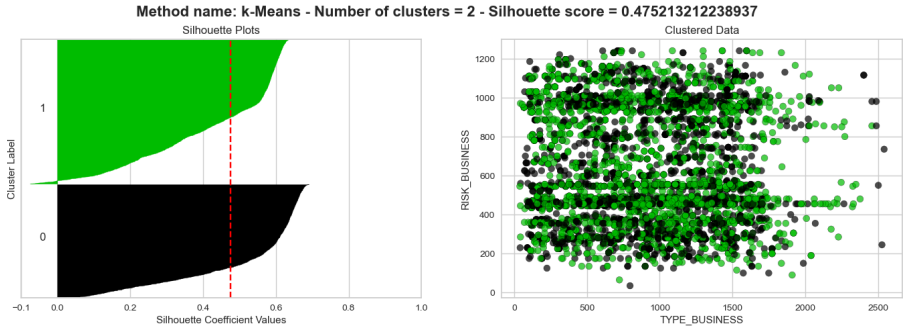

Figure 7.35: Silhouette Plot with k-Means algorithm - Feature subset: PA04 - K : 2

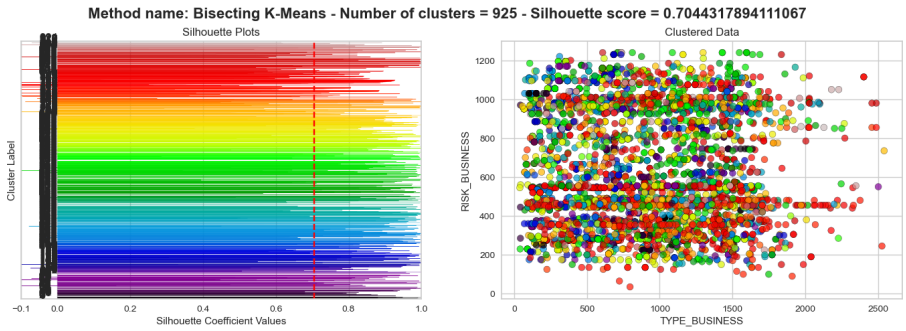

Figure 7.36: Silhouette Plot with Bisecting k-Means algorithm - Feature subset: PA01 - K : 925

A dataset is then described using a small number of exemplars, which are identified as those most representative of other samples. The messages sent between pairs represent the suitability for one sample to be the exemplar of the other, which is updated in response to the values from other pairs. This updating happens iteratively until convergence, at which point the final exemplars are chosen, and hence the final clustering is given

7.9.3 Affinity Propagation

Affinity Propagation creates clusters by sending messages between pairs of samples until convergence.
A dataset is then described using a small number of exemplars, which are identified as those most representative of other samples. The messages sent between pairs represent the suitability for a sample to be the exemplar for the other, which is then updated in response to the values from other pairs. This update happens iteratively until convergence, at which point the final exemplars are chosen, then the final clustering is given.
This algorithm does not require to specifiy the number of clusters.

The best results reached with Affinity Propagation algorithm is with the combination of the following parameters:

```
model=AffinityPropagation(random_state=5, damping=0.80, max_iter=10000)
```

The features subsets[5] used to compute the silhouette score and the results are:

- **PA01 : subset_Problem_A_UsefulFeatures**
  Estimated number of clusters obtained: 44 *Convergence Warning*
  Silhouette Score equals to: 0.7057490124670691
- **PA02 : subset_Problem_A_UsefulShortFeatures_onlyOriginalFeatures**
  Estimated number of clusters obtained: 44 *Convergence Warning*
  Silhouette Score equals to: 0.705749012467128
- **PA03 : subset_Problem_A_UsefulShortFeatures_withNewFeature**
  Estimated number of clusters obtained: 45
  Silhouette Score equals to: 0.6883323938607713
- **PA04 : subset_Problem_A_UsefulShortFeatures_Compact**
  Estimated number of clusters obtained: 71
  Silhouette Score equals to: 0.3358232601596728

Convergence Warning means that Affinity propagation algorithm did not converge in this case, this model may return degenerate cluster centers and labels.

[5]See the chapter 7.6.1

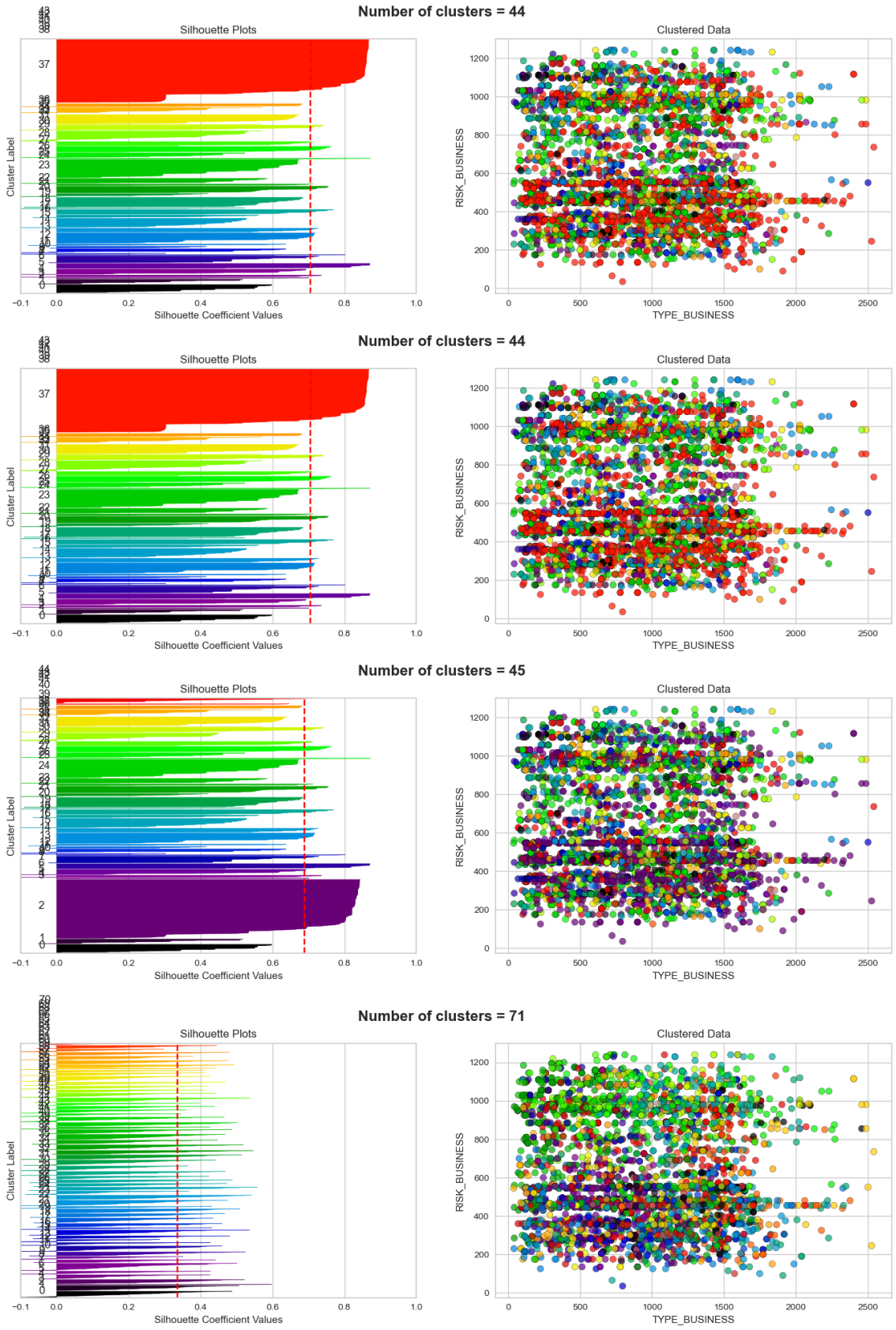

Figure 7.37: Silhouette analysis with Affinity Propagation algorithm (Subset: PA01, PA02, PA03, PA04).

7.9.4 Final Model

The main evaluation before realizing the final AI model is to decide the number of clusters as an input to the algorithm.
On Table 7.6 there is a summary of the highest silhouette score alongside with the correlated number of cluster for every subset features and algorithms tested.
During the analyses of different silhouette score charts it has been also identified local maximums but on the Table 7.6 are reported the global maximums, only.

Algorithm	Features Subset	The highest score	n. clusters
k-Means	PA01: subset_Problem_A_UsefulFeatures	**0.7104492843678244**	950
k-Means	PA02: subset_Problem_A_UsefulShortFeatures_onlyOriginalFeatures	0.7097356095596139	1000
k-Means	PA03: subset_Problem_A_UsefulShortFeatures_withNewFeature	0.7086974672882610	1075
k-Means	PA04: subset_Problem_A_UsefulShortFeatures_Compact	0.4752132122389370	2
Bisecting k-Means	PA01: subset_Problem_A_UsefulFeatures	0.7044317894111067	925
Bisecting k-Means	PA02: subset_Problem_A_UsefulShortFeatures_onlyOriginalFeatures	0.7044317926735026	925
Bisecting k-Means	PA03: subset_Problem_A_UsefulShortFeatures_withNewFeature	0.7044317927541638	925
Bisecting k-Means	PA04: subset_Problem_A_UsefulShortFeatures_Compact	0.4752018019872223	2
Affinity Propagation	PA01: subset_Problem_A_UsefulFeatures	0.7057490124670691	44
Affinity Propagation	PA02: subset_Problem_A_UsefulShortFeatures_onlyOriginalFeatures	0.705749012467128	44
Affinity Propagation	PA03: subset_Problem_A_UsefulShortFeatures_withNewFeature	0.6883323938607713	45
Affinity Propagation	PA04: subset_Problem_A_UsefulShortFeatures_Compact	0.3358232601596728	71

Table 7.6: Summary of silhouette score analysis on "k-Means", "Bisecting k-Means" and "Affinity Propagation" algorithms

Both the Algorithm and the Features Subset chosen are highlighted on Figure 7.6.
On Figure 7.38 is reported the analyses conduct as to identified the best value of K (number of cluster) starting from the Algorithm and the Features Subset chosen.

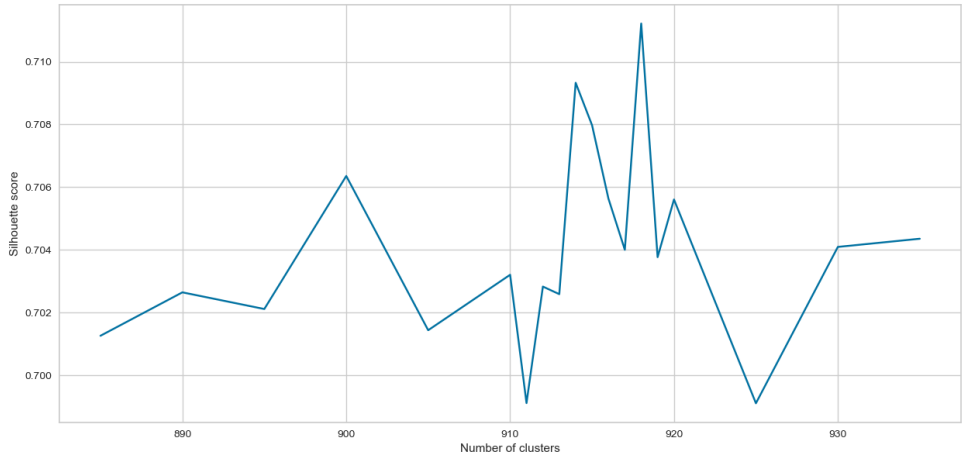

Figure 7.38: Focus on silhouette score analyses with k-Means algorithm from 885 to 935 clusters. Features subset: PA01

On Figure 7.38 is rappresented the values of the Silhouette scores compute, on features subset PA01, for differents number of clusters (from 885 to 935) with k-Means algorithm ran with the following parameters:

```
model=KMeans(n_clusters=k, init='k-means++', random_state=42, n_init=100)
```

The highest score (0.7112195465130113) was obtained by using 918 centers.

The final model which was realized to solve the business problem A was obtained by running the algorthm with a huge amount of initial random cluster (10.000 round). The algorithm ran with following parameters:

```
model=KMeans(n_clusters=k, init='k-means++', random_state=42, n_init=10000)
```

The final model which obtained a score of (0.7112195465130113) by using 918 centers. On Figure 7.39 it is rappresented the Silhouette plot related to the final model.

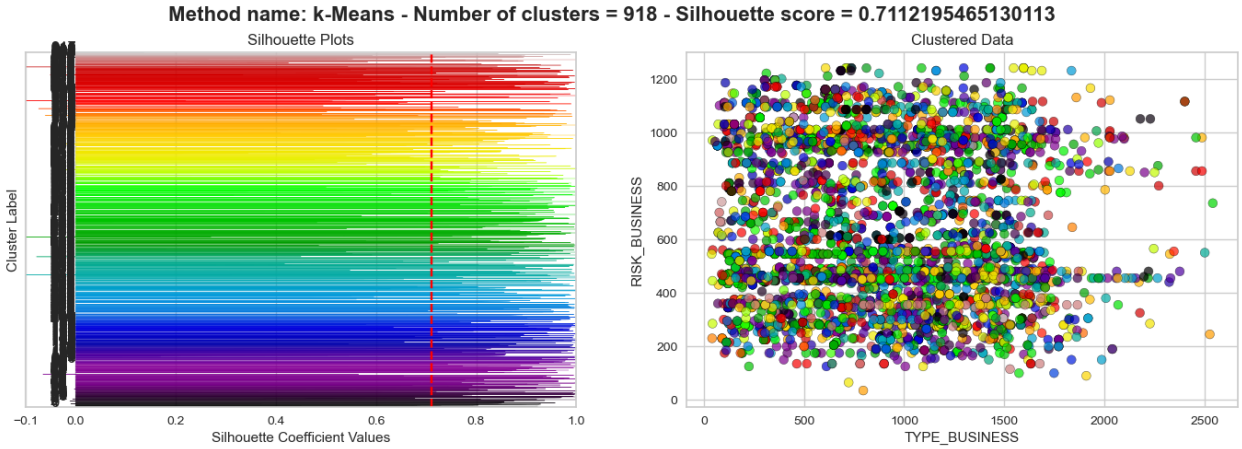

Figure 7.39: Silhouette plot related to the final model realized to solve the business problem A

7.10 Model to solve the Business Problem B

As already mentioned in the chapter 6, to solve the business problem B an Error Detecting and Tagging System is needed.
The theoretical "Error Detecting and Tagging System" was already introduced in the chapter 6 and schematized on Figure 6.3 and Figure 6.4 whilst the "Error Detecting and Tagging System" realized and implemented is reported in the Figure 7.40.
The schema shown on Figure 7.40 is focused on the input field named "Gross Premium 100", but it can be easily extended on others fields as defined in the Figure 5.2.

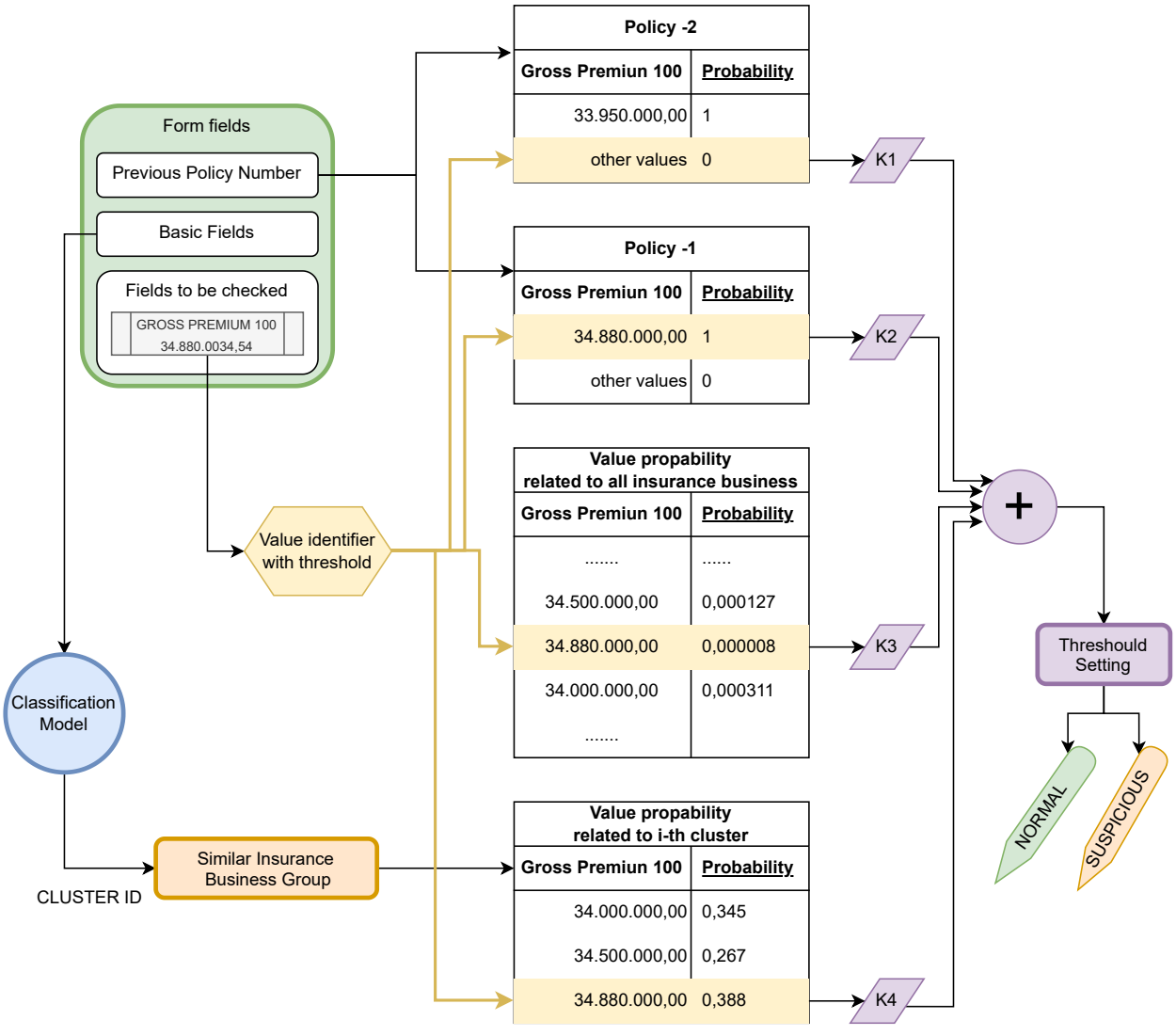

Figure 7.40: Real Schema Model B

The implemented version is quite different from the initial theoretical version presented in chapter 6 due to the following:

- the amount of data available (considering the number of insurance business for each cluster) is lower then expected. As a consequence the theoretical idea

to compute the value probability (for a input field) conditioned to other input fields values has changed;

- during the implementation phase, it emerged a very useful contribution of generic probability derived by all data, but not by the data in the same cluster,only;
- the majority of the insurance business in our dataset contain 2 policy renewals which means that it possible to consider 2 renewals as a maximum;

7.10.1 Schema description

The schema in the Figure 7.40 is described in the following bullet points:

- the green box represents the data typed by the user. It is possible to distinguish 3 types of input values:
  - "previous policy number" field that links the insurance business to the old policy (in the case of a policy renewal);
  - set of "basic fields" that will not be checked by the system
  - set of "fields to be checked" that will be checked by the system. In this schema we focus our attention on the input field named "Gross Premium 100";
- the blue box represents the classification model which classifies the data typed by the user into an insurance business cluster;
- thanks to the cluster ID return by the classification model, we are able to identify a set of similar insurance Business;
- the four tables represent the probability of the values referred to the GROSS PREMIUM 100 input field. These tables are pre-calculated during the system initialization phase.
  - the first two tables represent the probability values of the GROSS PREMIUM 100 input field on 2 past policies (past renewals);
  - the third table represents the probability values of the GROSS PREMIUM 100 input field considering all insurance business in our database;
  - the fourth table represents the probability values of the GROSS PREMIUM 100 input field considering all insurance business in our cluster indentified by the CLUSTER ID;
- the yellow box and arrows represent a system which starts from the value

typed by the user and identify a value probability on each table. This system uses a threshold to apply a tollerance on matching system between the GROSS PREMIUM 100 value typed by the user and the GROSS PREMIUM 100 values on tables;

- the purple boxes and arrows represent a system which:
  - starts from the 4 probability values derived from the four tables;
  - applies at each probability value a reduction (K_1 , K_2 , K_3 , K_4). This K-values correspond to weigths assign to different probability types. ($K_1 + K_2 + K_3 + K_4 = 1$);
  - aggregates the four weigth probabilities in one;
  - returns a "normal" tag if the probability is major of the threshould probability $P_{threashold}$ or a "suspicious" tag in others cases;

7.10.2 Classification model

The Classification Model represented in Figure 7.40 is a cluster system similar to the model used for solving the Business Problem A. The procedure followed to make the classification model is the same already presented in the chapter7.9.
The features subsets (presented in the chapter 7.6.1) used to build the model are:

- **PB01** ***subset_ UsefulFeatures_ Problem_ B***
- **PB02** ***subset_ UsefulShortFeatures_ onlyOriginalFeatures_ Problem_ B***
- **PB03** ***subset_ UsefulShortFeatures_ withNewFeatures_ Problem_ B***
- **PB04** ***subset_ UsefulShortFeatures_ Compact_ Problem_ B***

A silhouette score analysis is reported on Figure 7.41.

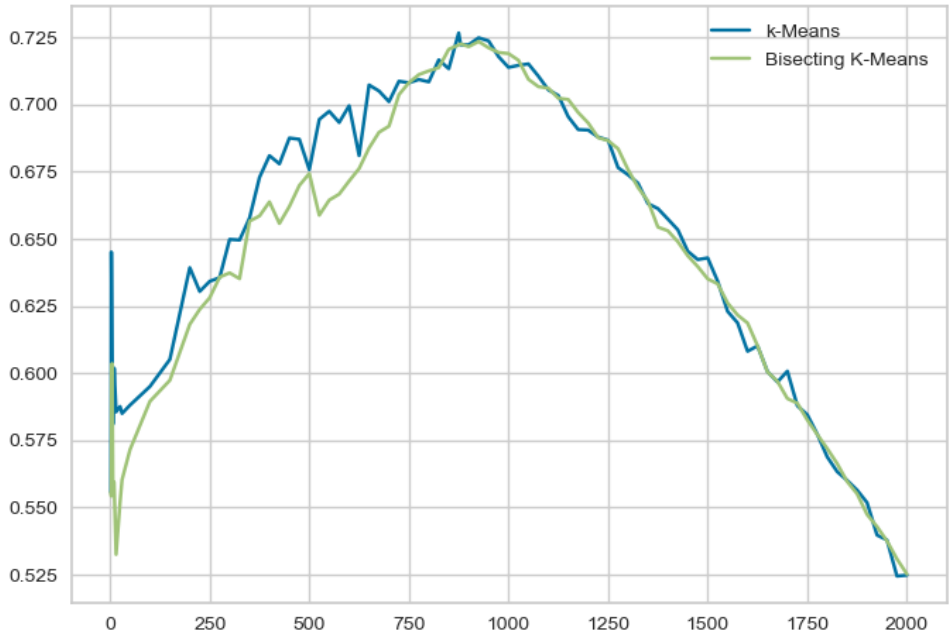

Figure 7.41: Silhouette score analysis

The final model realized to solve the business problem B was obtained by running the algorithm with a large amount of initial random clusters (10.000 round).
The algorithm ran with following parameters:

```
model=KMeans(n_clusters=k, init='k-means++', random_state=42, n_init=10000)
```

The final model obtained a score of 0.7251325447630233 using 880 centers.
On Figure 7.42 it is rappresented the Silhouette plot related to the final model.

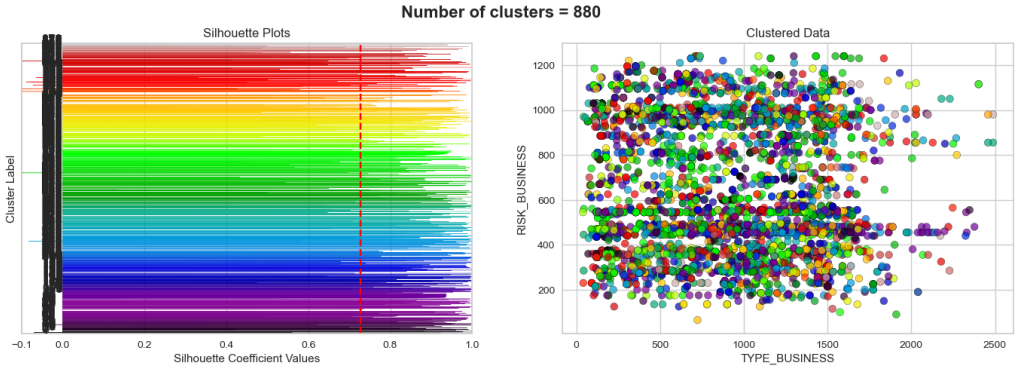

Figure 7.42: Silhouette plot related to the final model realized as to solve the business problem B

7.10.3 Final Model

The final model introduced in the Figure 7.40, has been implemented by:

- Creation of a classification model with Python's libraries as already said in the chapter 7.9;
- Creation a set of PL/SQL objects to implement the system report on Figure 7.40;

Oracle objects created/utilized to perform this task:

- creates 4 PL/SQL materialized views to compute the 4 tables introduced in the schema. This allows us to create a "tables system" that can be easly refreshed by forcing a materialized view refresh (only one PL/SQL istruction).
- creates a PL/SQL function to identify a probability on each table.
- creates PL/SQL function to aggregate the four probabilities and return the tag value: "Normal" or "Suspicious".

Model tuning

The tuning phase to set the costants values of the system K_1 , K_2 , K_3 , K_4 and $P_{threashold}$, was performed by using the cases test produced by the business unit under the evaluation system phase introduced in the next chapter.
By Using the tests cases provided, different values constant combinations were tested as to maxify the system' result which means : maxify the number of correct tags "normal" or "suspicious" returns by the system.

Chapter 8

Evaluate the models

According to what has just been described in the on chapter 7, especially:

- in the chapter 7.9.4 about "Low number of clusters", "Is overfitting a problem in unsupervised learning?" and "Final Model";
- in chapter 7.10.3 about "Final Model";

the latest evaluation of the AI Models related to business problems solving goes to the Business Unit.
In this Chapter the evaluation process made by the insurers as to evaluate the performed activity will be described, then.

8.1 Model evaluation related to the Business case A

During the making model phase the student provided the business unit reference with different models.
For each and every model the business unit made two different evaluations:

- **Clusters Density Analysis**
  Analysis about how the insurance business is divided.
- **Manual clusters evaluation by the Business Unit**
  Analysis of some clusters as to check if the related insurance business are similiar whilst following the business unit exspectations.

The business unit gave a mark (from A to D) for each and every model analyzed. The model with the highest mark was choosen, then. On Figure 8.1 is presented the Cluster Density plot related to the final model realized to solve the business problem A.

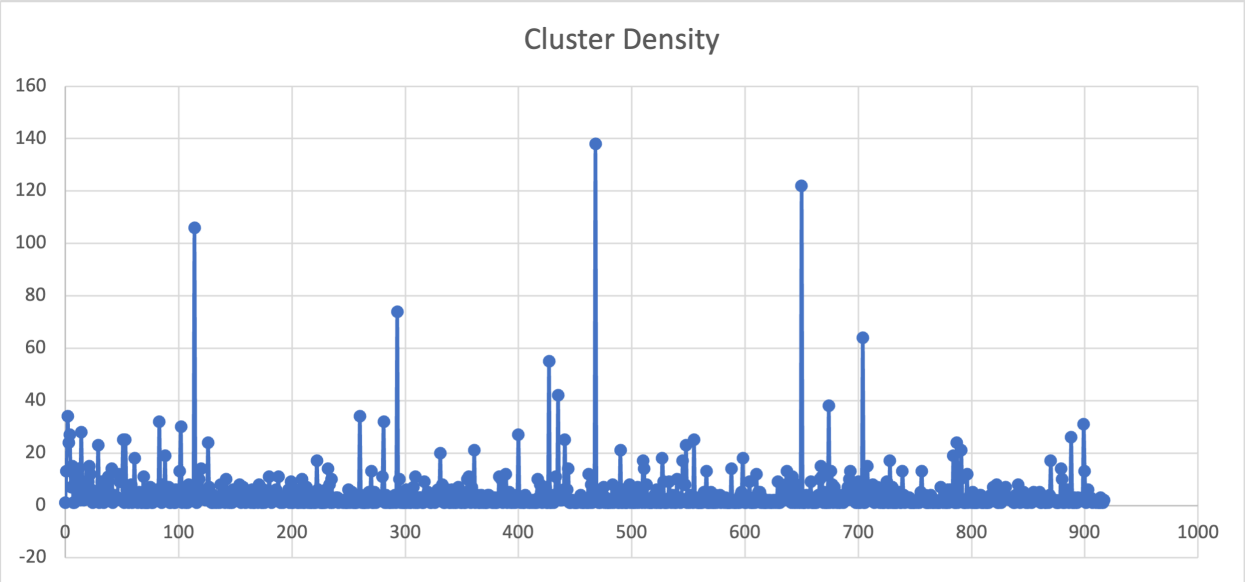

Figure 8.1: Cluster Density plot related to the final model realized to solve the business problem A

8.2 Model evaluation related to the Business case B

During the initial analysis phase of the problem B the business unit provided a set of samples to summarize the typical data entry errors made by the employees. 4 type of errors have been identified on the base of this set of examples:

1. "single typed digit error" such as "12395,067" instead of "12345,067";
2. "decimal integer separation error" such as "123450,67" instead of "12345,067";
3. "insurance policy reference value error" in other words the employeestypes a value amount correlated to a different insurance policy;
4. "field's input error" in other words the employee types a value amount in a different field within the correct insurance policy;

The type of errors identified were used to realize an algorithm in order to produce random cases tests. These cases tests were then used to evaluate the final model B. The cases tests set created was used during the model tunning's phase. On this basis, we can conclude that the best solution was obtained with those variables values:

$$K_1 = 0,13 \;\text{-}\; K_2 = 0,46 \;\text{-}\; K_3 = 0,05 \;\text{-}\; K_4 = 0,36 \;\text{-}\; P_{threashold} = 0,78$$

The model created and set with these variables values perform 73% of right tests cases. A correct case test means that the model return the right tag: "normal" or

"suspicious".
By Observing the variables values identified it is possible to deduct how the model works: during the tagging phase the model defines a priority order to the sub-models as follows:

1. sub-model related to the previous policy (years -1)
2. sub-model related to the previous policy (years -2)
3. sub-model related to the similar insure policies
4. sub-model related to the entire dataset of policies

Chapter 9

The web application

In this chapter will be presented the web application realization phase.

9.1 Technology architecture

The technology architecture to realize this application is introduced with the Figure 9.1 and it can be summarized as follows:

- Angular CLI: 14.2.6 to realize the front-end part.
  Angular is a platform and framework for building single-page client applications by using HTML and TypeScript
- Spring Boot to realize the back-end part of the application.
  Java Spring Boot (Spring Boot) is a tool that allows to develop web application and microservices with Java Spring Framework both faster and easier.
- Oracle database.

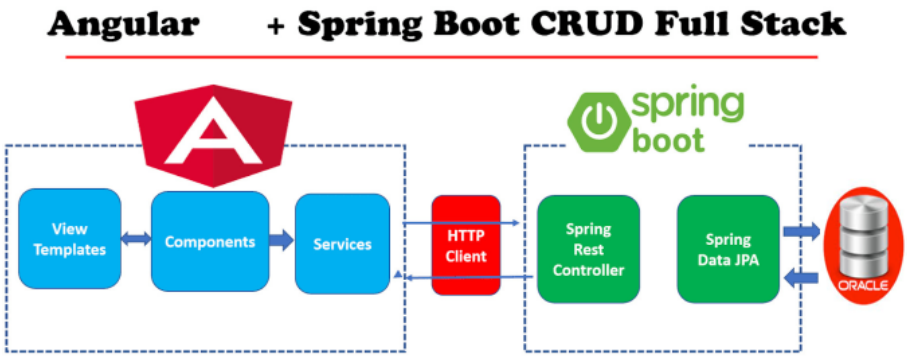

Figure 9.1: Application technology architecture

9.2 User Interface

Starting from the mockups realized during the initial business analysis phase and presented in the Chapter 3;real HTML web pages have been designed and coded. The relevant pages have been reported in the Figures 9.2, 9.3, 9.4.
The head information about a specific business is shows both in the Figure 9.2 and the Figure 9.3.

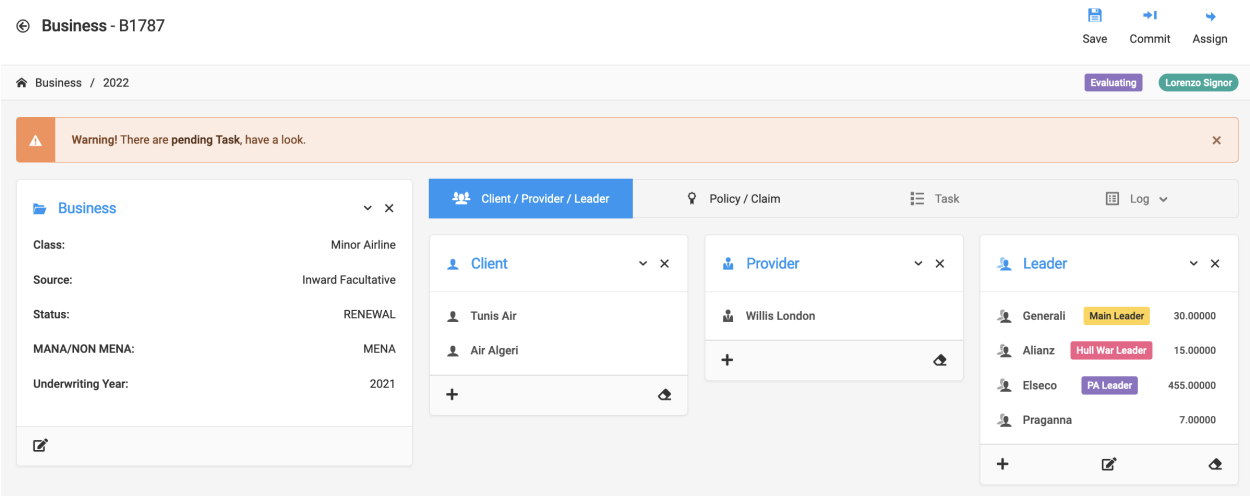

Figure 9.2: User interface: Head information about a specific business

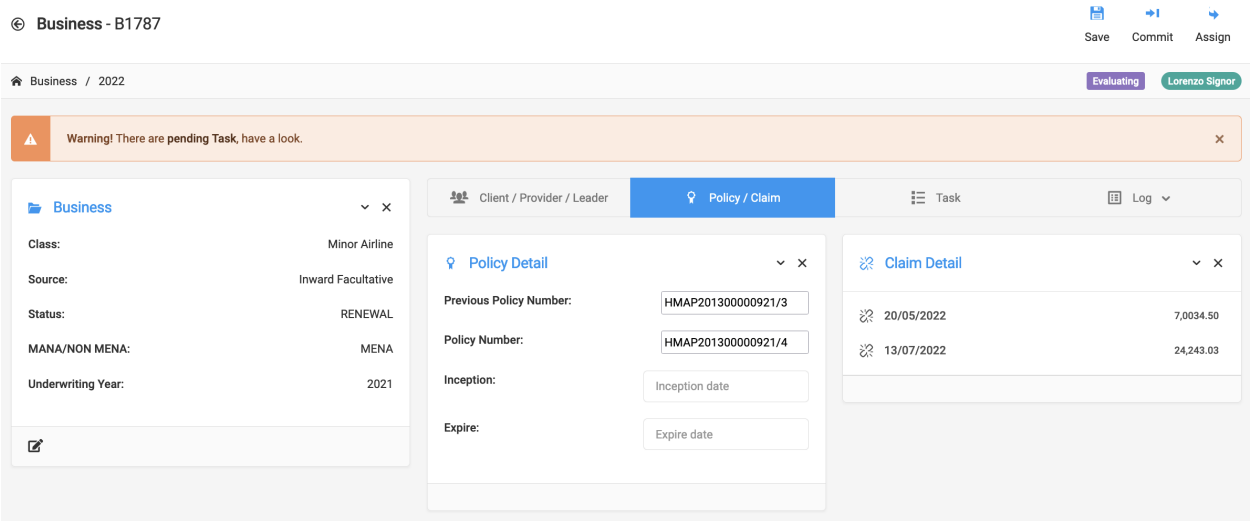

Figure 9.3: User interface: Policy and Claims Section

The main information about a specific business are shown in the Figure 9.4. In such a Figure, you will find some green checks which are the consequence of the input fields validation made by the Model B in order to solve the Business Problem B.

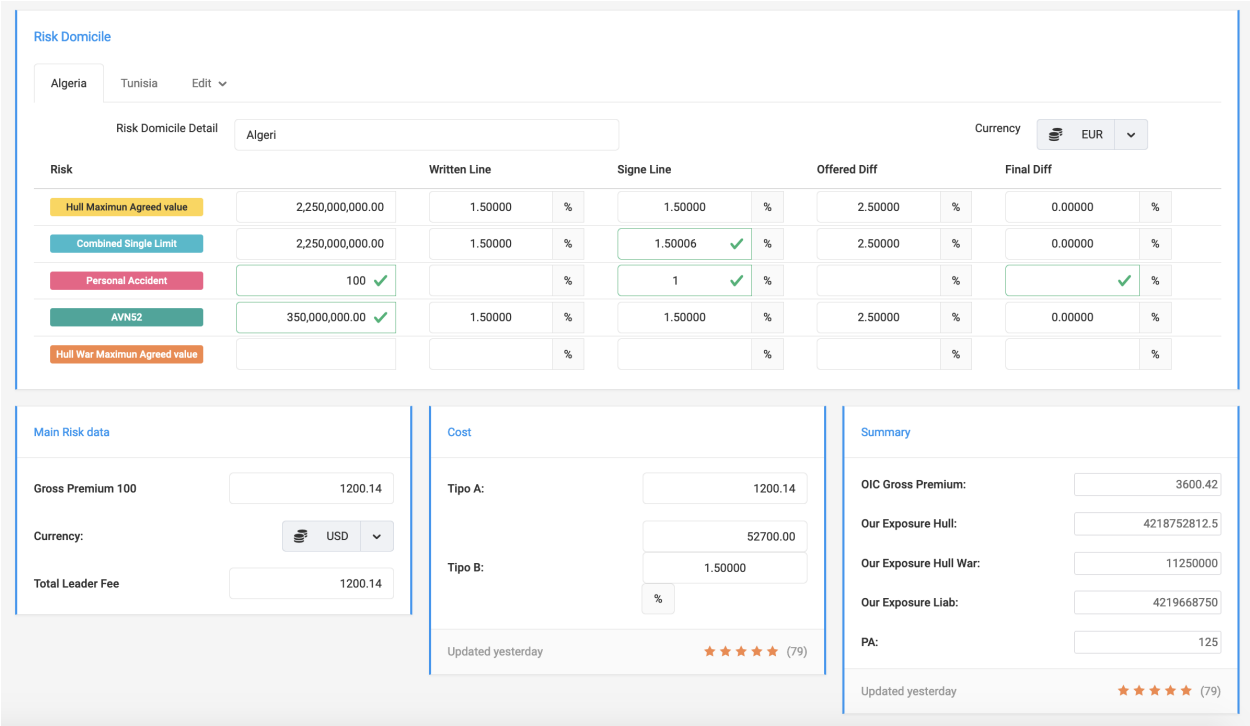

Figure 9.4: User interface: Main information about a specific business

9.3 Spring Boot back-end services

The main spring boot backend services which have been implemented can be summarized as follows:

- Some services to save the data typed by the user related to an insurance business
- A service which use the Model A
- A service which use the Model B

9.3.1 Service which use the Model A

This service receives an ID which identifies a specific insurance business as an input (ID_BUSINESS), then returns a list of past insurance business. This service uses the Model realized to solve the Business Problem A. The operations provided with this service are:

1. The java service reads all necessary data related to the insurance business identified by the ID which has been received as an input. This service is taken by the database, of course;
2. The java service calls the Python code which implements the "AI Model - Problem A" to clusterize the insured business under evaluation;

3. The java service saves the CLUSTER_ID (obtained by the python service call) on the database;

4. The java service reads from the database a list of insurance business related to the same cluster ID, then returns the list to the front-end side application;

9.3.2 Service which uses the Model B

This service receives as an input:

- the ID which identifies a specific insurance business (ID_BUSINESS);
- the name of the field which needs to be checked;
- the value field amount typed by the user;

and returns a "normal" or "suspicious" tag. This service uses the Model realized to solve the Business Problem B. The operations made by this service are the following:

1. The java service calls the PL/SQL Function providing the inputs already described, then receives the tag value;

2. The java service returns the tag value to the front-end side application;

Realization note

The web application realization phase was not fully completed by the student. In fact, all necessary spring boot back-end services to support the front-end pages except the 2 services described were realized by the IT team of the company.

Conclusion

The thesis work performed represents an example of a complete flow of an information system creation that uses AI in order to implement a sequence of advanced functionalities. The analyzed case study allowed us to address all the project phases, which we may summarize as follows:

- the initial analysis and collection of requirements;
- the identification of the data lake;
- the analysis and cleaning of the data;
- the creation of theoretical models to solve the business problems expressed by the business unit;
- the complete implementation of the hypothesized models;
- the creation of the CRM application that uses the models;

Throughout the project phases, various challenges emerged, which were tackled through careful analysis and selection of the best resolution/management approach among those identified.
The two problems presented by the business unit made it possible to focus attention on two aspects:

the creation of a "pure AI model" for solving the first business problem: "a tool capable of supporting the insurance underwriter during the evaluation phase of a new insurance risk to be underwritten".
The developed clustering model, based on a unsupervised and inductive learning, it is able to provide a set of past insurance business (already evaluated and underwritten) that are highly similar to the one currently being considered for underwriting, thus assisting the insurance underwriter during this phase.
Considering the nature of the examined data (Aerospace insurance risks), the results obtained using the unsupervised AI model have been excellent compared to the results obtained by developing more traditional models.

the creation of a "hybrid model" to solve the second business problem: "a system

which allows both to identify and tag data entry errors made by users". Unlike the previous model, this model combines a probabilistic and an AI model together to provide a feedback to the user whether the entered data is correct or potentially incorrect/suspicious. This system is able to detect erroneous input values that looks like normal statistically; but abnormal in the current context.

The main criticality encountered was the collection of data, their migration towards a structured data system and their cleaning. These initial activities which have been essential to start developing our models, have proved to be onerous, due to the fact that the current data management is carried out "manually" by using various Excel files.

The development of theoretical models to solve various business cases has highlighted the vastness of AI models/algorithms currently available and the difficulty in identifying a type that was concretely usable, with good performance and scalable over time.

The creation of AI models using real-case data have also highlighted the importance of the work performed during the data cleaning and preparation phase, as well as during the feature engineering phase.

Whilst in literature, the stages of an AI model development are presented as sequential, IE "Data Preparation" and "Model Development"; in reality, an iteractive cycle of activities was followed during our work, with the mentioned phases being repeated multiple times as to refine the models on a step by step basis. The cycles of activities (data preparation and model training) came to an end when the model's performance met the specified requirements.

The results obtained met the expectations of the business unit which then decided to bring the entire system created with both models implemented into production.

Bibliography

[1] Sukoon Insurance Company.
Homepage company web site.
https://www.sukoon.com.

[2] Dubai Financial Market (DFM).
Company profile web page.
Dubai Financial Market web site https://www.dfm.ae/issuers/listed-securities/securities/company-profile-page?id=OIC.

[3] PMI Project Management Institute.
How to manage a milestone, or managing by deliverables.
https://www.pmi.org/learning/library/manage-milestone-managing-deliverables-10437.

[4] Cambridge Business English Dictionary.
Mock-up definition.
https://dictionary.cambridge.org/dictionary/english/mock-up.

[5] Seobility.net.
Mock-up definition.
https://www.seobility.net/en/wiki/Mockup.

[6] Oracle.
Oracle Database Express Edition distribution.
https://www.oracle.com/database/technologies/appdev/xe.html.

[7] Git Hub.
Colima Project.
https://github.com/abiosoft/colima#installation.

[8] hub.docker.com.
Oracle Database Express Edition Docker.
https://hub.docker.com/r/gvenzl/oracle-xe.

[9] Oracle.
Oracle SQL Developer.
https://www.oracle.com/database/sqldeveloper/technologies/download/.

[10] Oracle.
Oracle SQL Format.
https://docs.oracle.com/database/121/SQLRF/sql_elements004.htm.

[11] scikit-learn web-site.
scikit-learn : Machine Learning in Python.
`https://scikit-learn.org/stable/index.html`.

[12] scikit-learn web-site.
Clustering algorithm on scikit-learn library.
`https://scikit-learn.org/stable/modules/clustering.html`.

[13] KRISTINA P. SINAGA **and** MIIN-SHEN YANG.
?Unsupervised K-Means Clustering Algorithm**?**
IEEE Access, Digital Object Identifier: 10.1109/ACCESS.2020.2988796 (April 5, 2020).

[14] ZHONG-BIN ZHANG.
?Research of Error Data Detection Algorithm Based on Rules**?**
IEEE Access, Digital Object Identifier: 978-1-61284-486-2/11 (2011).

[15] Feby a Vinisha.
?Study on Misssing Values and OutlierDetection in Concurrence with Data Quality Enhancement for Efficient data Processing**?**
IEEE Xplore Part Number CFP22P17-ART (2022).

[16] Larry A. Dunning.
?Signle and Double Length Error Detecting Decimal Codes**?**
ISIT 2002, Lausanne, Switzerland (June, 2002).

[17] Joseph M. Hellerstein Kuang Chen.
?Usher: Improving Data Quality with Dynamnic Forms**?**
IEEE Transactions on knowledge and data engineering, Vol. 23, NO. 8 (August 2011).

[18] W. Wong.
?Bayesian network anomaly pattern detection for disease outbreaks**?**
Machine learning-internationl workshop the conference (2003).

[19] Yuan An Yuan Ling.
?An Error Detecting and Tagging Framework for Reducing Data Entry Errors in Electronic Medical Records System**?**
IEEE Access, Digital Object Identifier: 978-1-61284-486-2/11 (2013).

[20] Joseph M. Hellerstein Kuang Chen.
?Designing Adaptive Feedback for Improving Data Entry Accuracy**?**
Proceedings of the 23th annual ACM symposium on User interface software and technology (2010).

[21] J.M. Hellerstein.
?Quantitative data cleaning for large databases**?**
United Nations Economic Commission for Europe (UNECE) (2008).