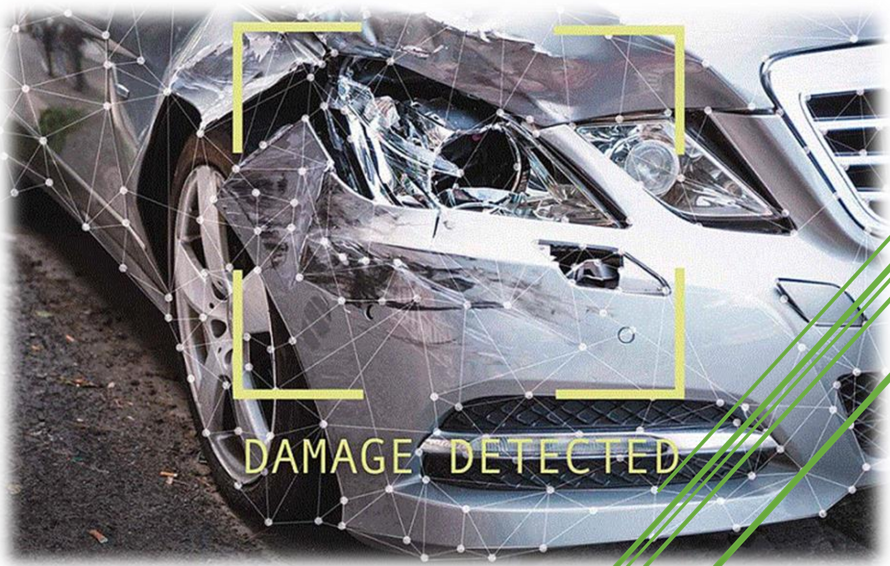
AUTOMATIC DAMAGES RECOGNITION

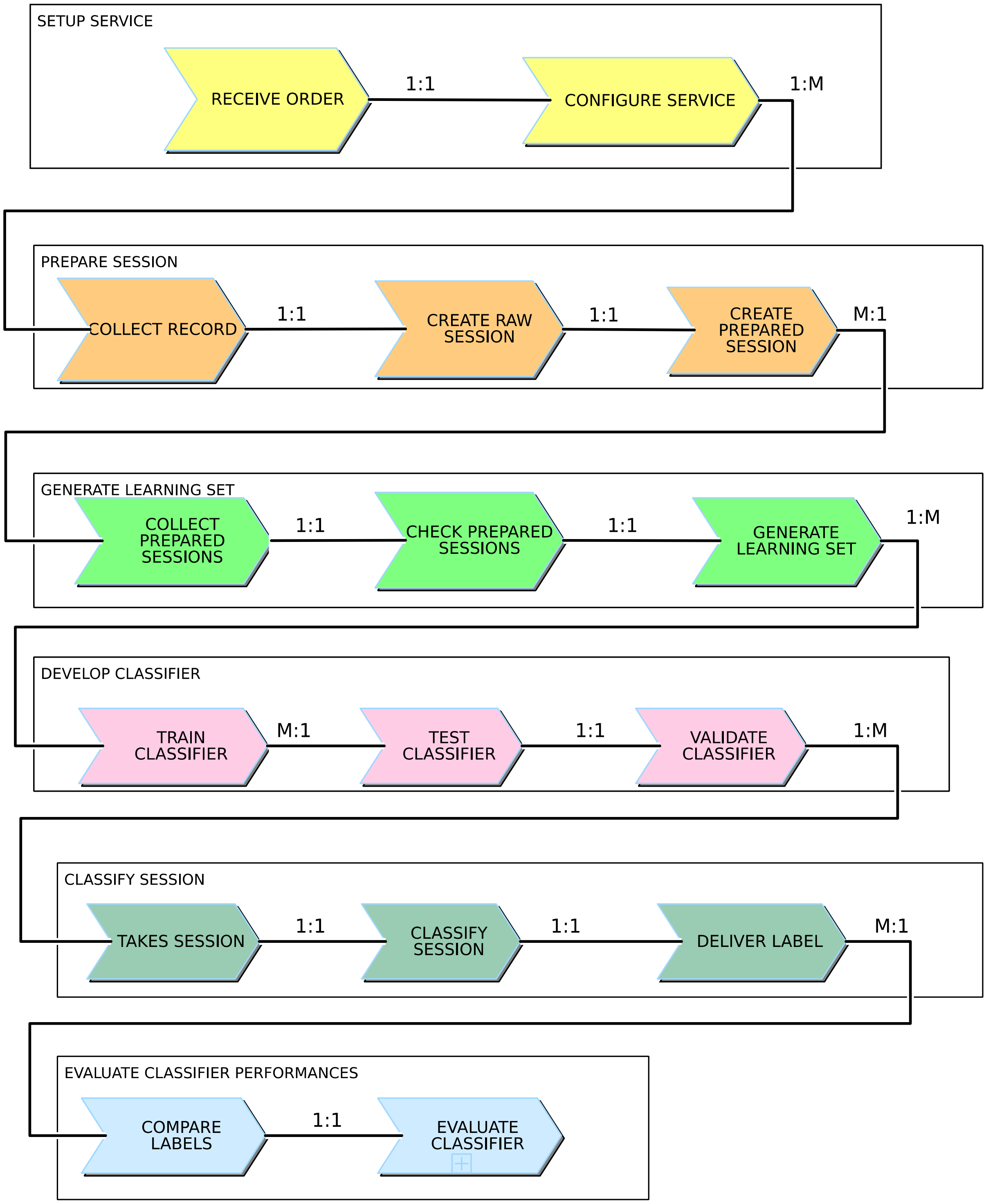
Giulio Bello, Giulio Capecchi, Federico Frati, Jacopo Niccolai



PROCESS MINING AND INTELLIGENCE PROJECT
A.A 2023/2024

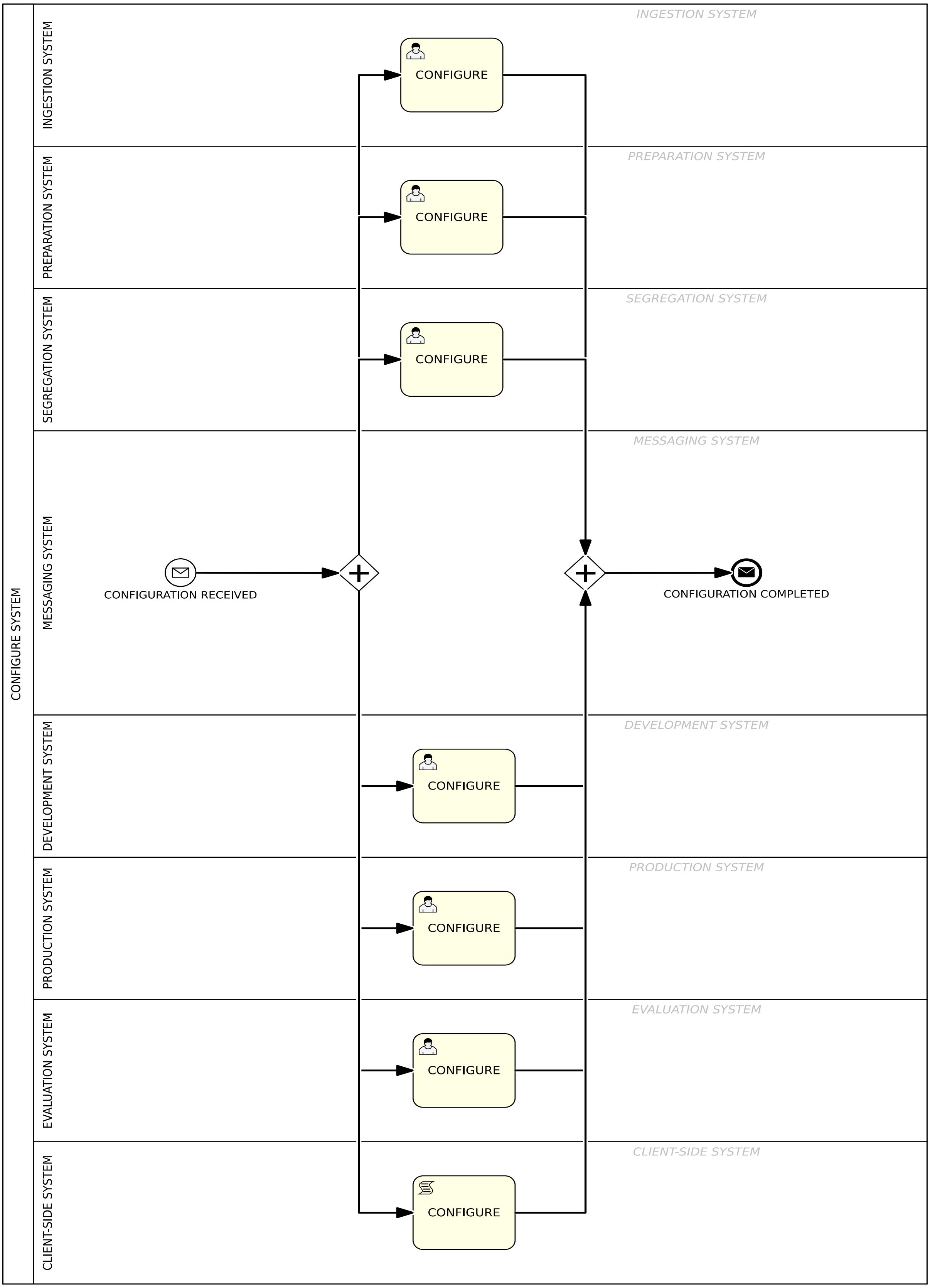
1 SOMMARIO

2 LANDSCAPE (BELLO, CAPECCHI, FRATI, NICCOLAI)



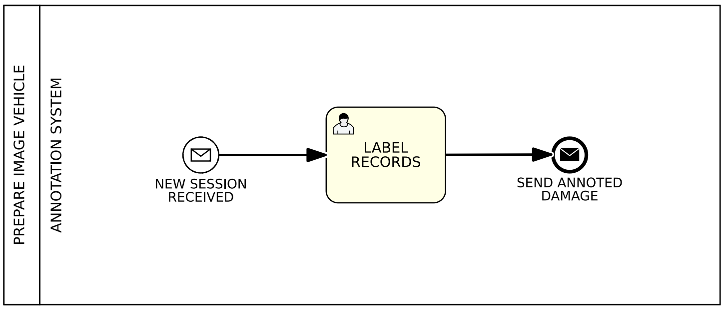
3 BPMN DIAGRAMS

3.1 CONFIGURE SYSTEM (BELLO, CAPECCHI, FRATI, NICCOLAI)



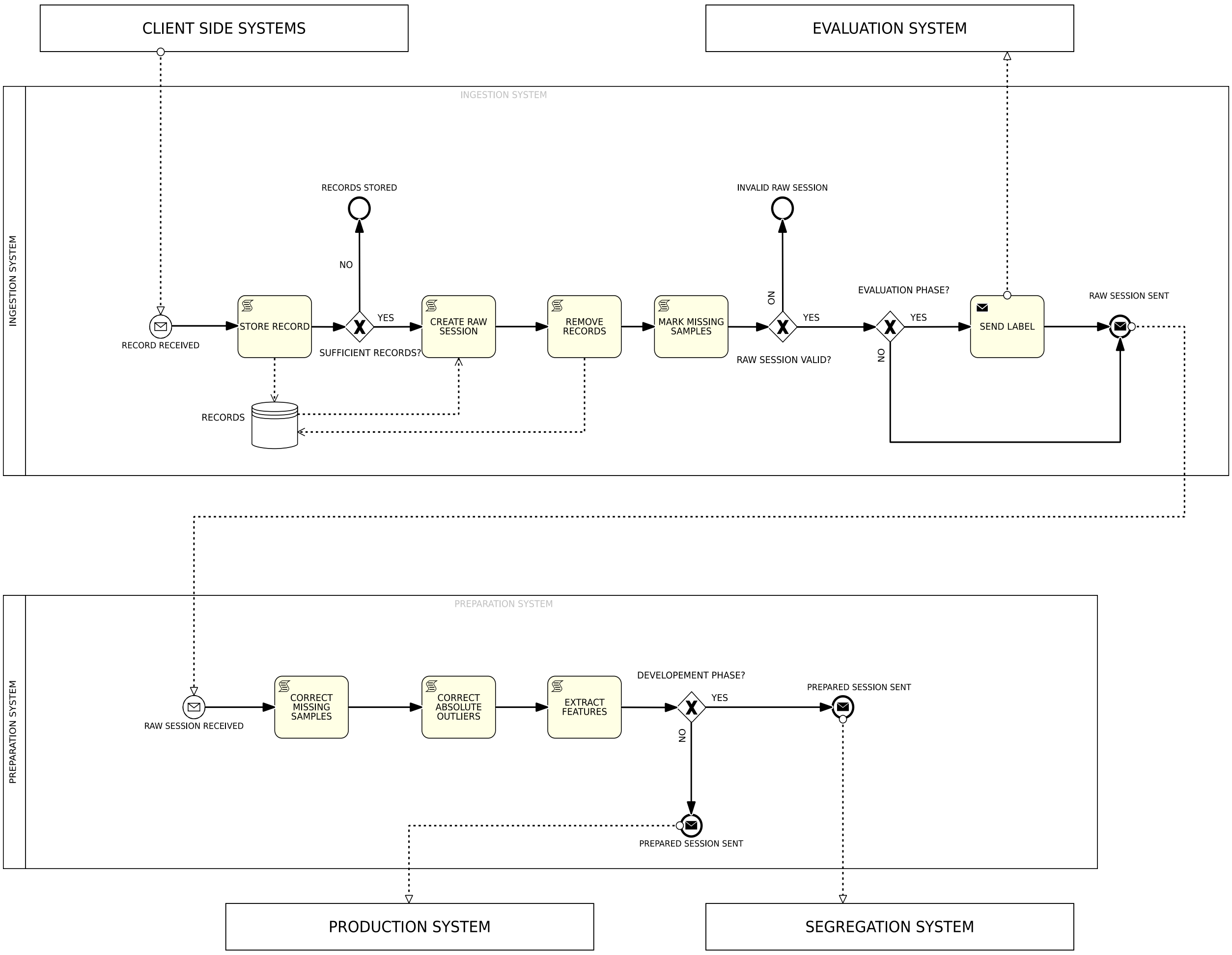
All systems need to be configurated before the usage. Each of these is configurated by a human actor (later we will see which actor it is) except by the *Client-Side System* that is configured with a script task.

3.2 ANNOTATION SYSTEM (FRATI)



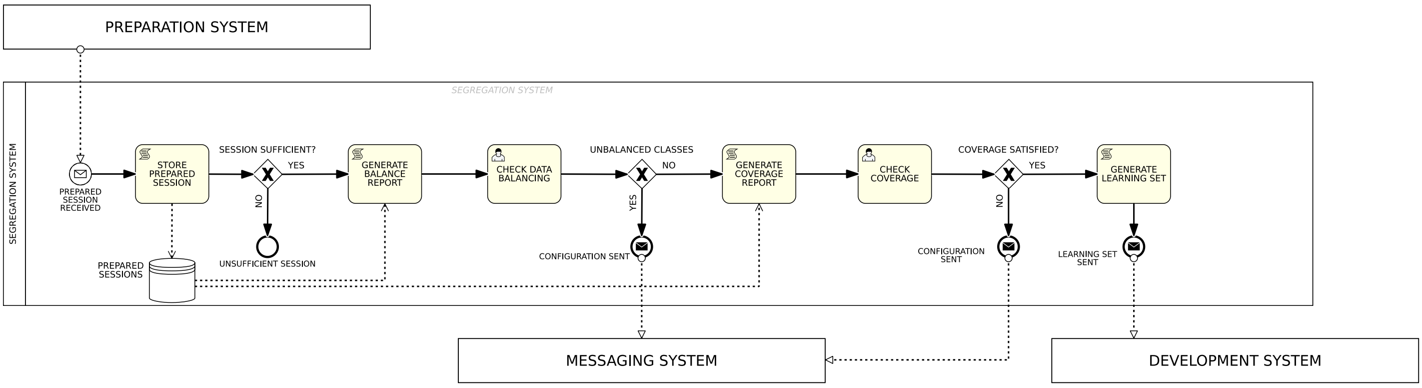
The intervention of the automobile damage appraiser in some moments is needed to have an evaluation of the damages. The start event is the reception of a session to classify, in fact the appraiser needs all the photos to make a general and complete evaluation. His evaluation is mandatory for development and evaluation phases.

3.3 PREPARE SESSION (CAPECCHI)



The *records* are received in the ingestion system; they make up the "*raw session"*, which is a group of them. The session is then analyzed by marking its missing samples. These are managed in the preparation system, where they're corrected together with the absolute outliers. From this preprocessed object, the features are then extracted, and the obtained result is called "*prepared session*", which is then sent to the right system depending on which phase the factory is in.

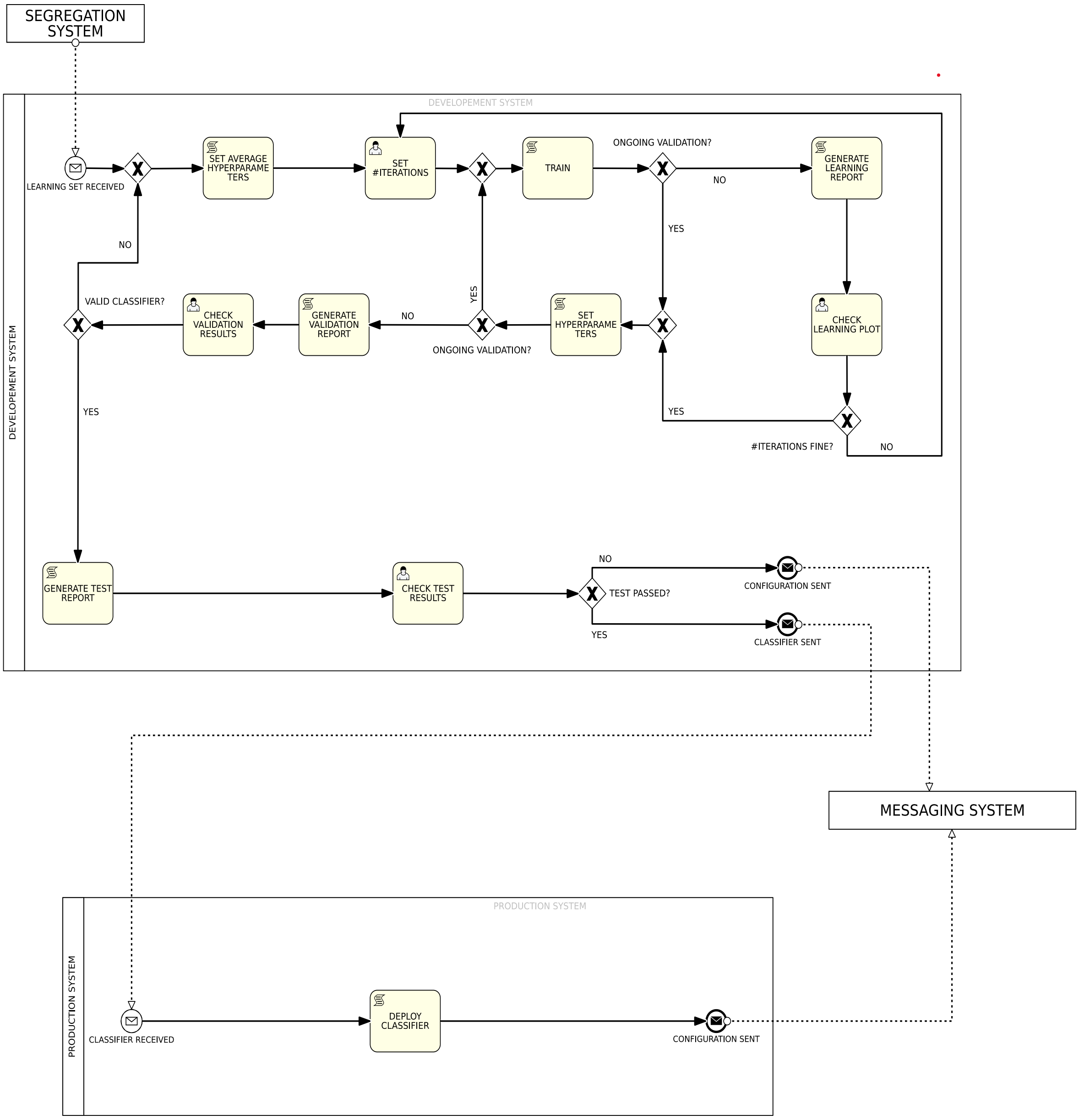
3.4 GENERATE LEARNING SET (FRATI)



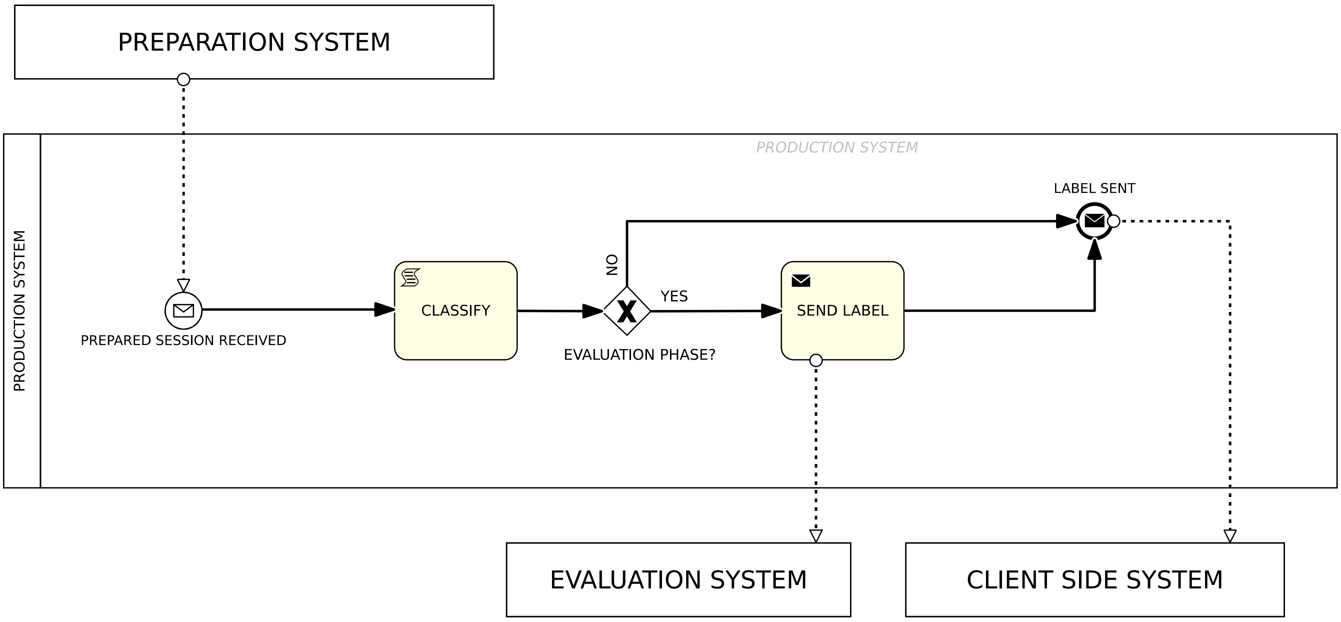
The start event is the reception of a prepared session from the Preparation System. This prepared session is stored and then is evaluated. If it is sufficient, some checks are performed: balancing and coverage check. These are made by human actors with specific knowledge on data analysis. In case of bad results of these checks, a message is sent to the messaging system to notify the problem found and to signal which classes or features are the critic ones.

If no problems are detected, the learning set generated is sent to the development system.

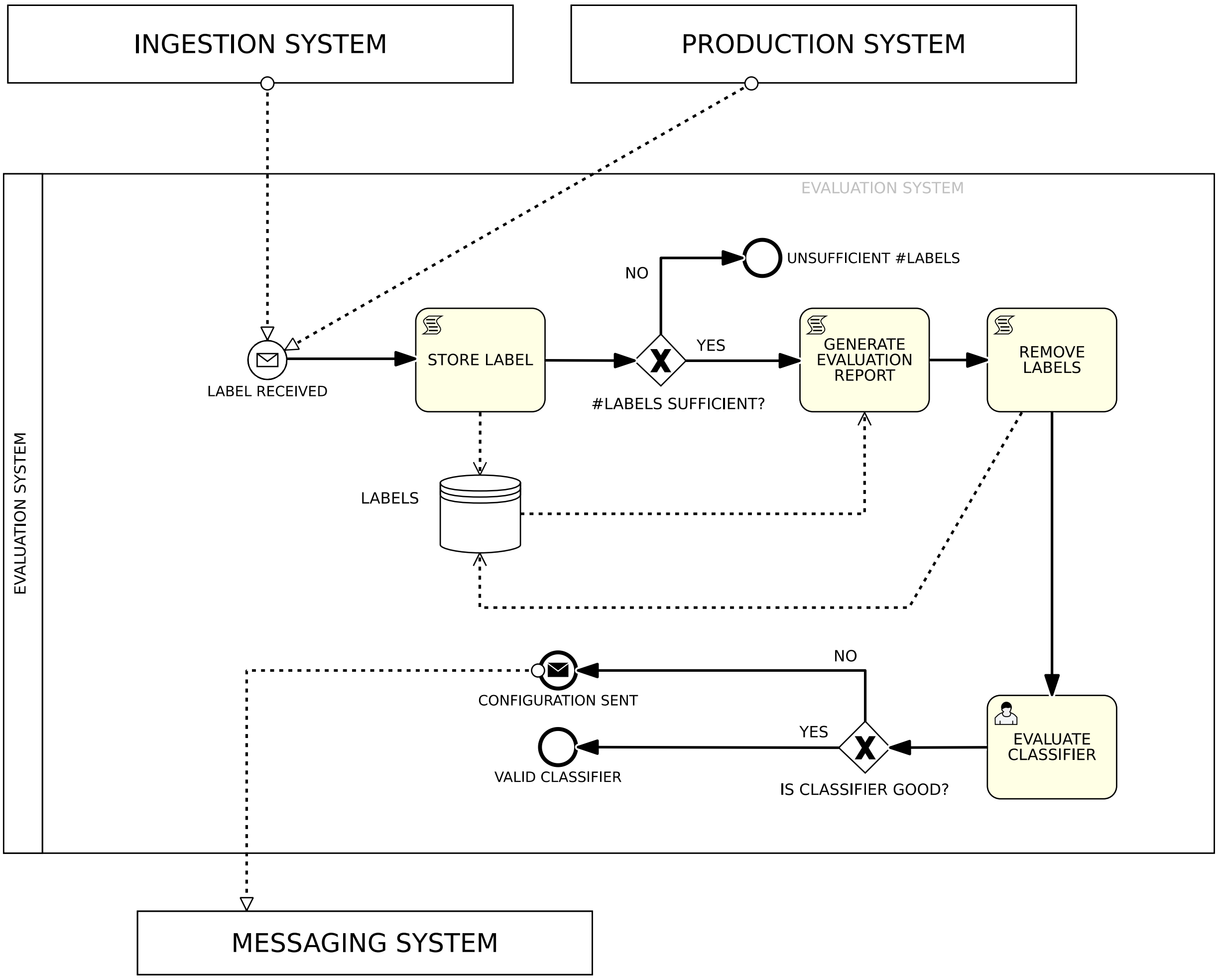
3.5 DEVELOP CLASSIFIER (BELLO)



3.6 CLASSIFY SESSION (NICCOLAI)



3.7 EVALUATE CLASSIFIER PERFORMANCE (NICCOLAI)



4 TASK LEVEL MODELING AND TASK COST

4.1 SALARY (BELLO, CAPECCHI, FRATI, NICCOLAI)

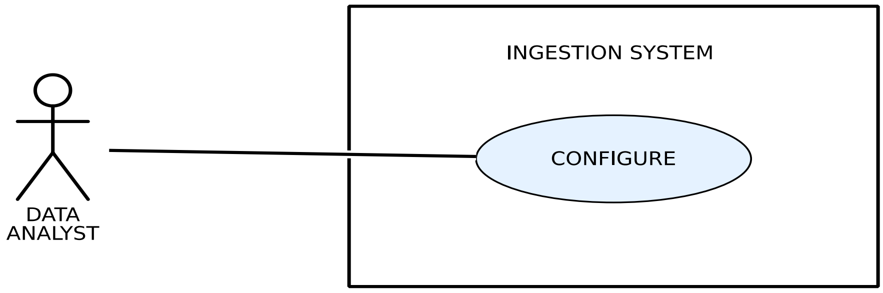
Job Role	Mean Salary (€)	Normalized Salary	Source
Data Analyst	48 000	1	https://www.prospects.ac.uk/job-profiles/data-analyst
Machine Learning Engineer	120 000	2.5	https://www.prospects.ac.uk/job-profiles/machine-learning-engineer
Automobile Damage Appraiser	62 000	1.3	https://www.erieri.com/salary/job/automobile-damage-appraiser/germany/heidelberg

- **Data Analyst:** examines and interprets data to extract meaningful insights, facilitating informed decision-making within organizations.
- **Machine Learning Engineer:** develops and deploys machine learning algorithms and models, enabling computers to learn and perform tasks without explicit programming.
- **Automobile Damage Appraiser:** assesses vehicle damage, estimating repair costs for insurance purposes, and ensures adherence to safety and quality standards in the automotive industry.

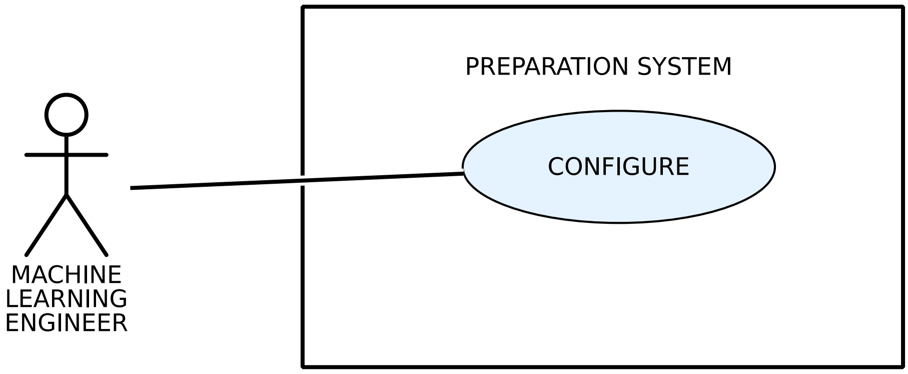
The main source we used to determine salaries for the roles we identified, reports the money in pounds and it does not have the amount corresponding for the automobile damage appraiser. So, we searched other sources where we finally found it but expressed in euros respect to the salary average for Germany. To have an appropriate relationship with the other salaries we decide to compare the amount with the salary average for Great Britain and by doing the proportion we obtained the reported value.

4.2 TASK LEVEL MODELING (BELLO, CAPECCHI, FRATI, NICCOLAI)

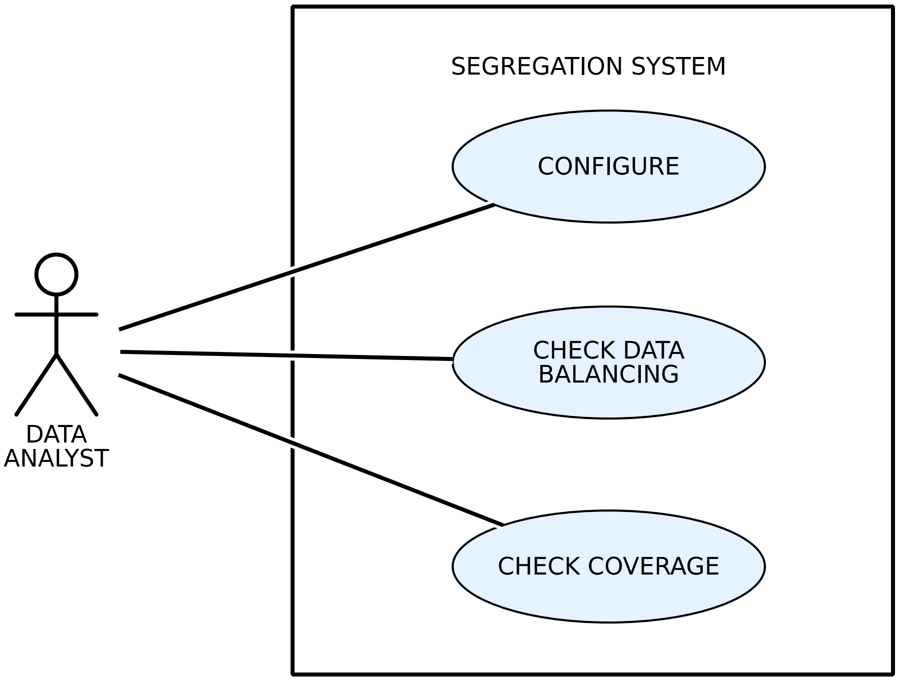
4.2.1 INGESTION SYSTEM



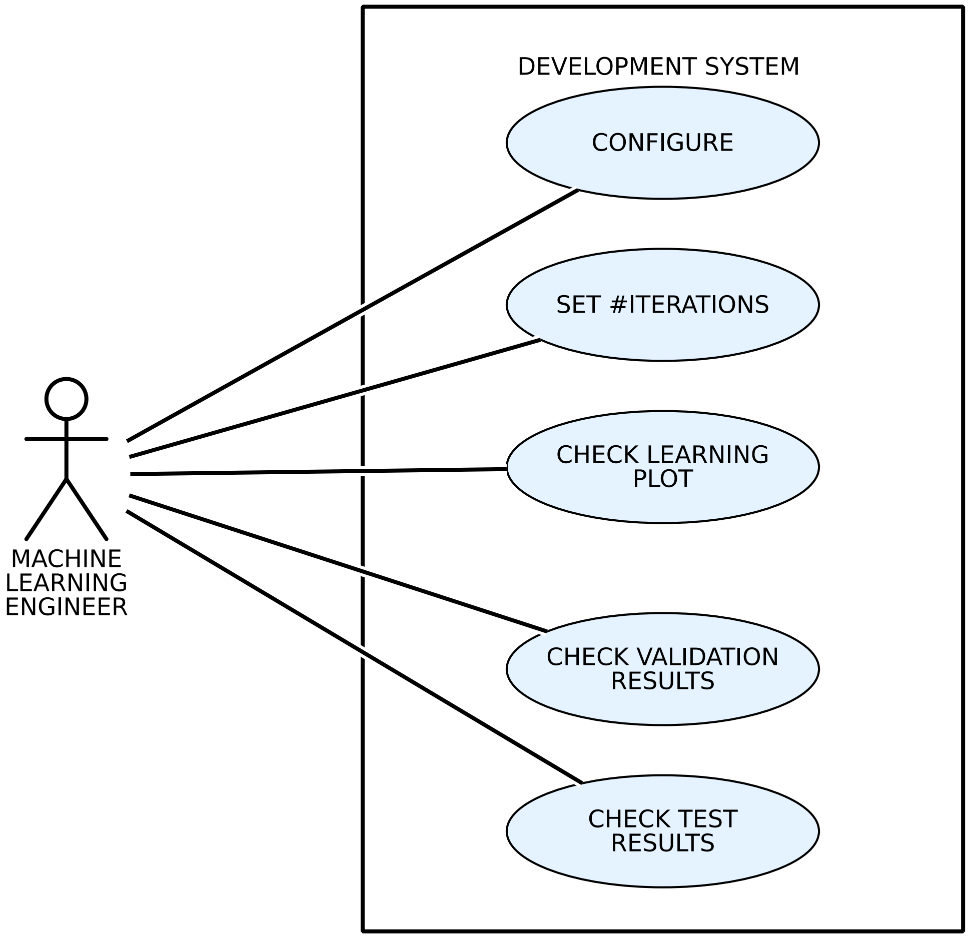
4.2.2 PREPARATION SYSTEM



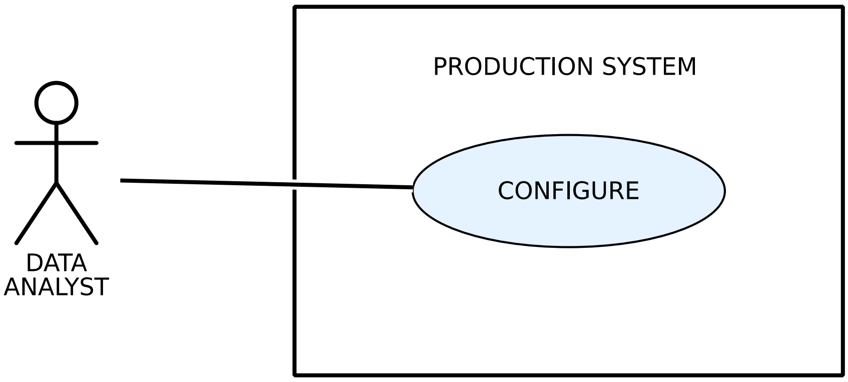
4.2.3 SEGREGATION SYSTEM



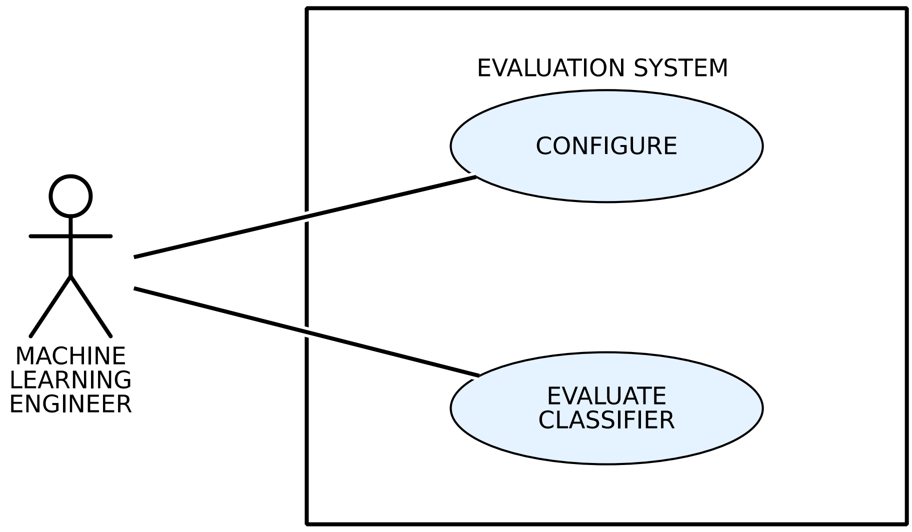
4.2.4 DEVELOPMENT SYSTEM



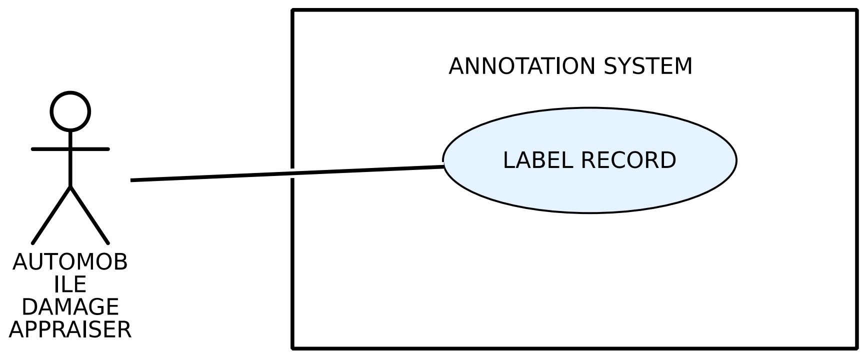
4.2.5 PRODUCTION SYSTEM



4.2.6 EVALUATION SYSTEM



4.2.7 ANNOTATION SYSTEM



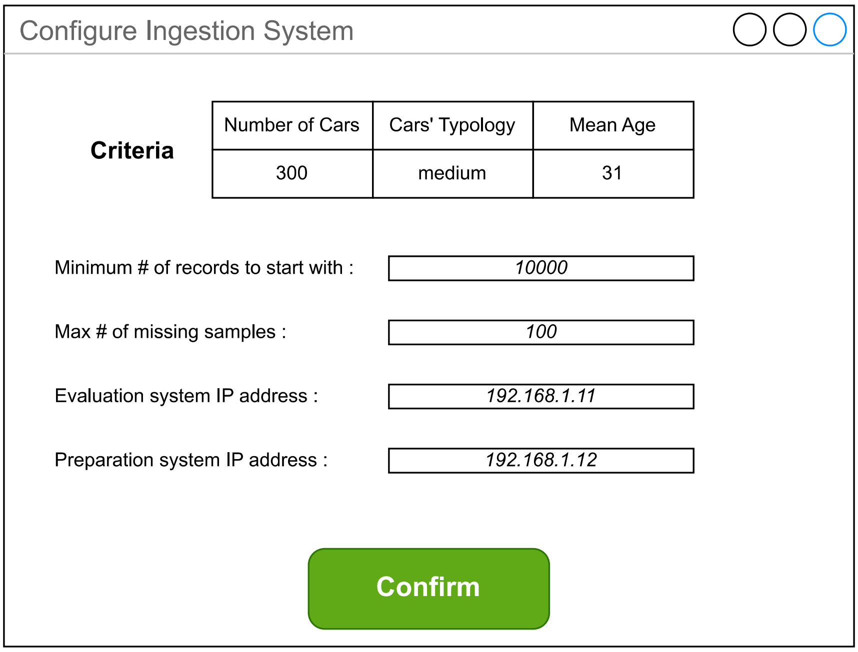
4.3 CONFIGURE

Note: In red we find a further multiplication (other than the standard *occurrence * cognitive effort * user cost*), because the users in that cases are obliged to use their past experiences to evaluate these *multicriterial decisions*; we choose three the number of these criteria, because we imaged to have three types of diverse factories that the users must think of in these cases. These are:

- Number of auto vehicles belonging to the fleet of cars
- The cars' typology (*low* for older vehicles, *medium* for standard/family cars, and *high* for expensive ones)
- Mean age of the clients

4.3.1 INGESTION SYSTEM (CAPECCHI)

USER is a *Data Analyst.*

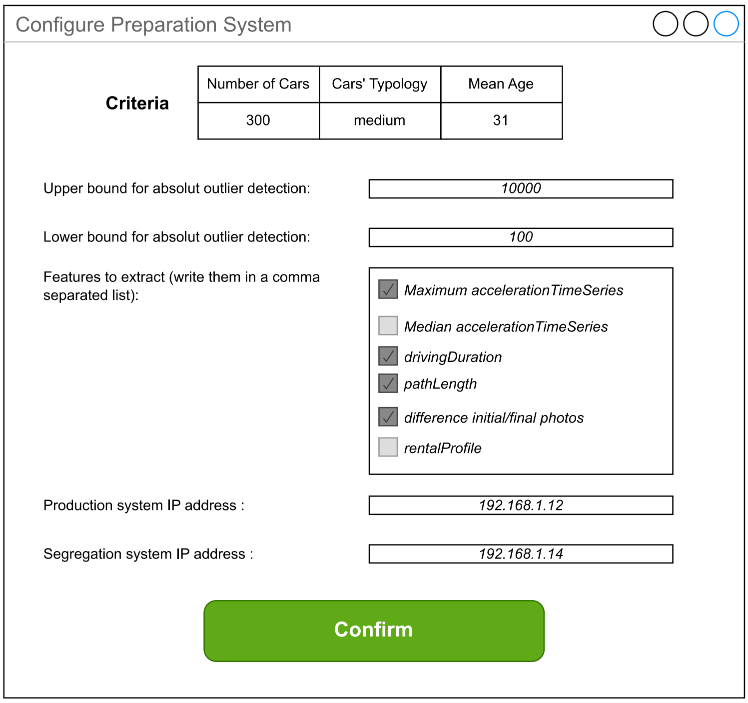


STEP	COST CALCULATION	STEP COST
1 USER opens ingestion system configuration page	1 x 1 x 1	1
2 SYSTEM shows ingestion system configuration page		
3 USER sets minimum # of records to start with	1 x 4 x 1 x 3	12
4 USER sets maximum # of missing samples	1 x 4 x 1 x 3	12
5 USER sets the IP address of Evaluation system	1 x 1 x 1	1
5 USER sets the IP address of Preparation system	1 x 1 x 1	1
6 SYSTEM enables the "CONFIRM" button		
7 USER presses confirm	1 x 1 x 1	1
8 USER closes configuration page	1 x 1 x 1	1
	HUMAN TASKS COST	**29**

In red we find the multicriterial decision for the Data Analyst, that must use his expertise in this case to decide those parameters, basing on the type of factory he is working with.

4.3.2 PREPARATION SYSTEM (CAPECCHI)

USER is a *Machine Learning Engineer*

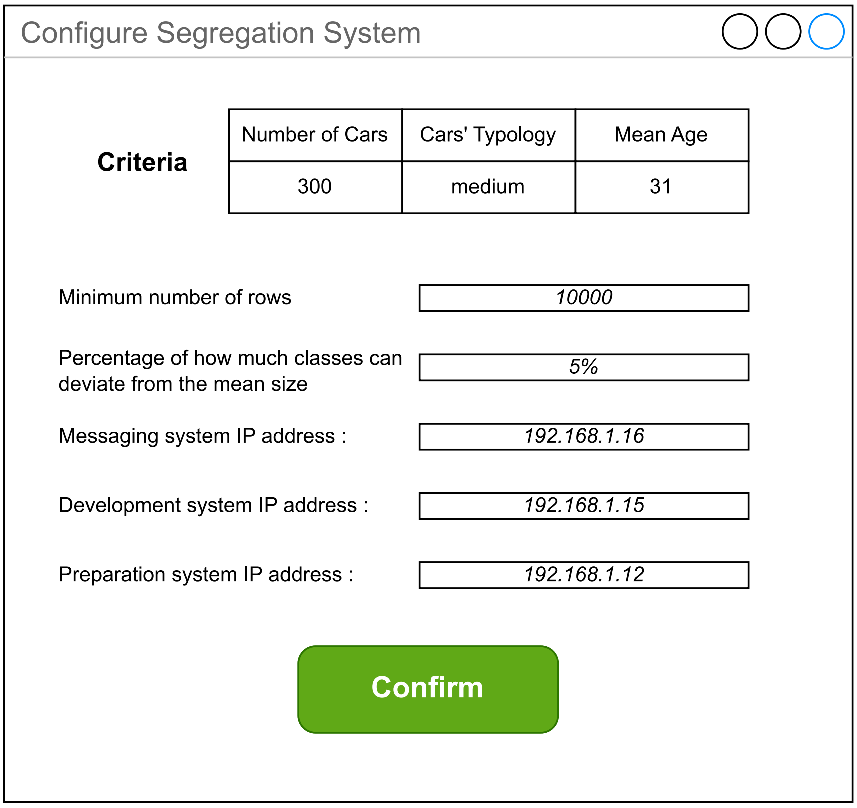


STEP	COST CALCULATION	STEP COST
1 USER opens preparation system configuration page	1 x 1 x 2.5	2.5
2 SYSTEM shows preparation system configuration page		
3 USER sets upper bound for absolut outlier detection	1 x 4 x 2.5 x 3	30
4 USER sets lower bound for absolut outlier detection	1 x 4 x 2.5 x 3	30
5 USER sets features to extract	6 x 4 x 2.5	60
5 USER sets the IP address of Production system	1 x 1 x 2.5	2.5
6 USER sets IP address of Segregation system	1 x 1 x 2.5	2.5
7 SYSTEM enables the "CONFIRM" button		
8 USER presses confirm	1 x 1 x 2.5	2.5
8 USER closes data balancing form	1 x 1 x 2.5	2.5
	HUMAN TASKS COST	**132.5**

In red we find the multicriterial decision for the Machine Learning Engineer, that must use his expertise in this case to decide the range for the absolute outlier detection, basing on the type of factory he is working with.

4.3.3 SEGREGATION SYSTEM (FRATI)

USER is a *Data Analyst*

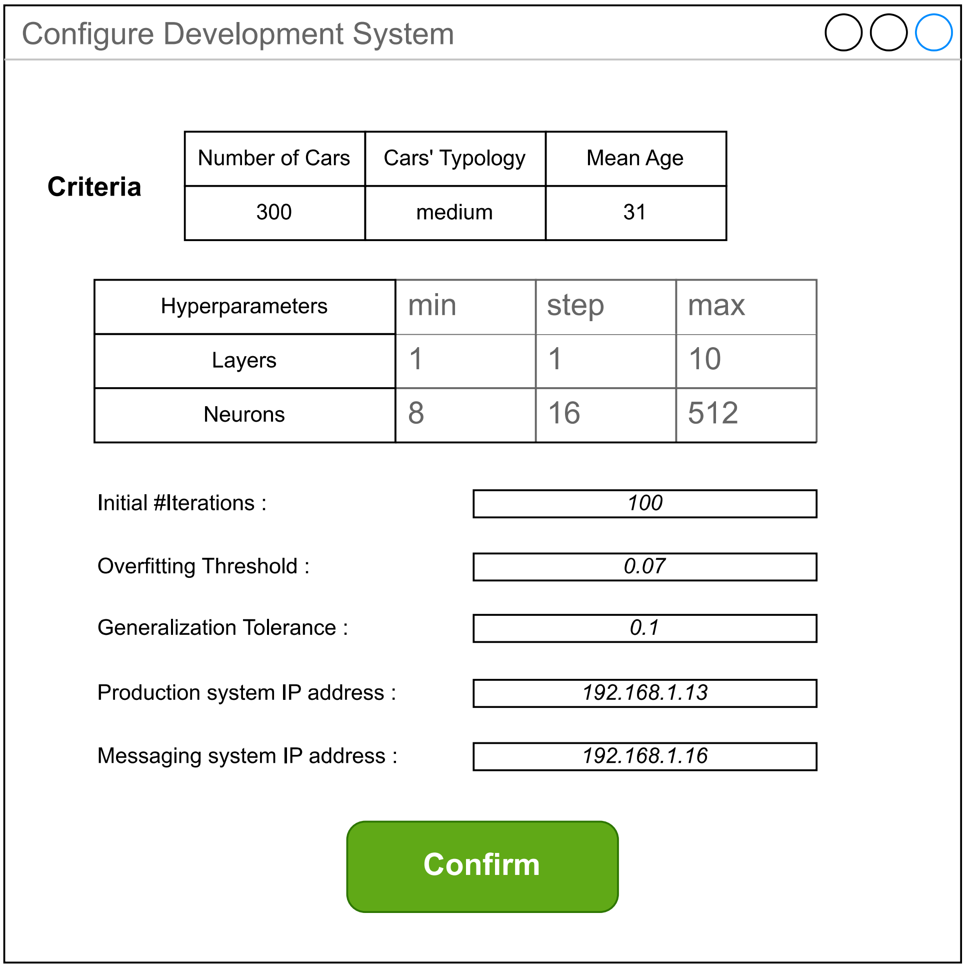


STEP	COST CALCULATION	STEP COST
1 USER open Configure Segregation System window	1 x 1 x 1	1
2 SYSTEM shows the window without setted parameters		
3 USER checks the 3 criteria	3 x 4 x 1	12
4. User sets the minimum required number of rows	1 x 4 x 1 x 3	12
5. USER sets the threshold percentage for the allowed deviation from the mean	1 x 4 x 1 x 3	12
6 USER insert Messaging system's IP	1 x 1 x 1	1
7 USER insert Development system's IP	1 x 1 x 1	1
8 USER insert Preparation system's IP	1 x 1 x 1	1
9 USER press Confirm	1 x 1 x 1	1
10 USER closes Configure Segregation System window	1 x 1 x 1	1
	HUMAN TASK COST	**42**

The user is a Data Analyst because these steps need a big knowledge on data analysis. There are three activities that need a cognitive effort of level 4: The first one is the checking of the three criteria that establish the kind of factory we are speaking about, while the others are relative to parameters of the characteristics of the desired learning set.

4.3.4 DEVELOPMENT SYSTEM (BELLO)

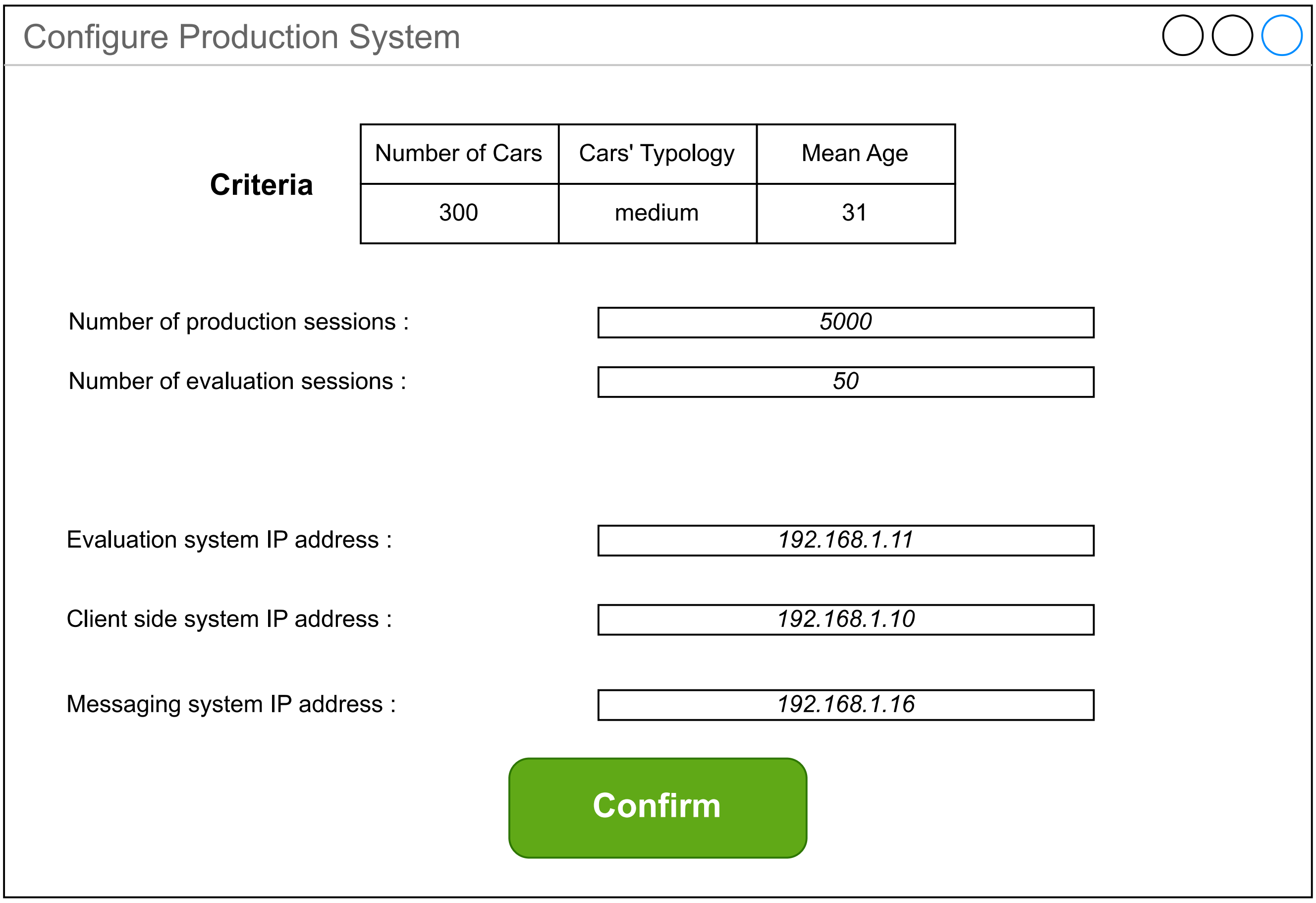
USER is a *Machine Learning Engineer*



STEP	COST CALCULATION	STEP COST
1 USER opens development system configuration window	1 x 1 x 2.5	2.5
2 SYSTEM shows development system configuration window		
3 USER checks the 3 criteria	3 x 4 x 2.5	30
4 USER sets Hyperparameters	6 x 4 x 2.5 x 3	180
5 USER sets initial number of iterations	1 x 4 x 2.5 x 3	30
6 USER sets Overfitting Threshold	1 x 4 x 2.5 x 3	30
7 USER sets Generalization Tolerance	1 x 4 x 2.5 x 3	30
8 USER sets Messaging System IP address	1 x 1 x 2.5	2.5
9 USER set Production System IP address	1 x 1 x 2.5	2.5
10 SYSTEM enables "CONFIRM" button		
11 USER clicks on "CONFIRM" button	1 x 1 x 2.5	2.5
	HUMAN TASKS COST	**310**

4.3.5 PRODUCTION SYSTEM (NICCOLAI)

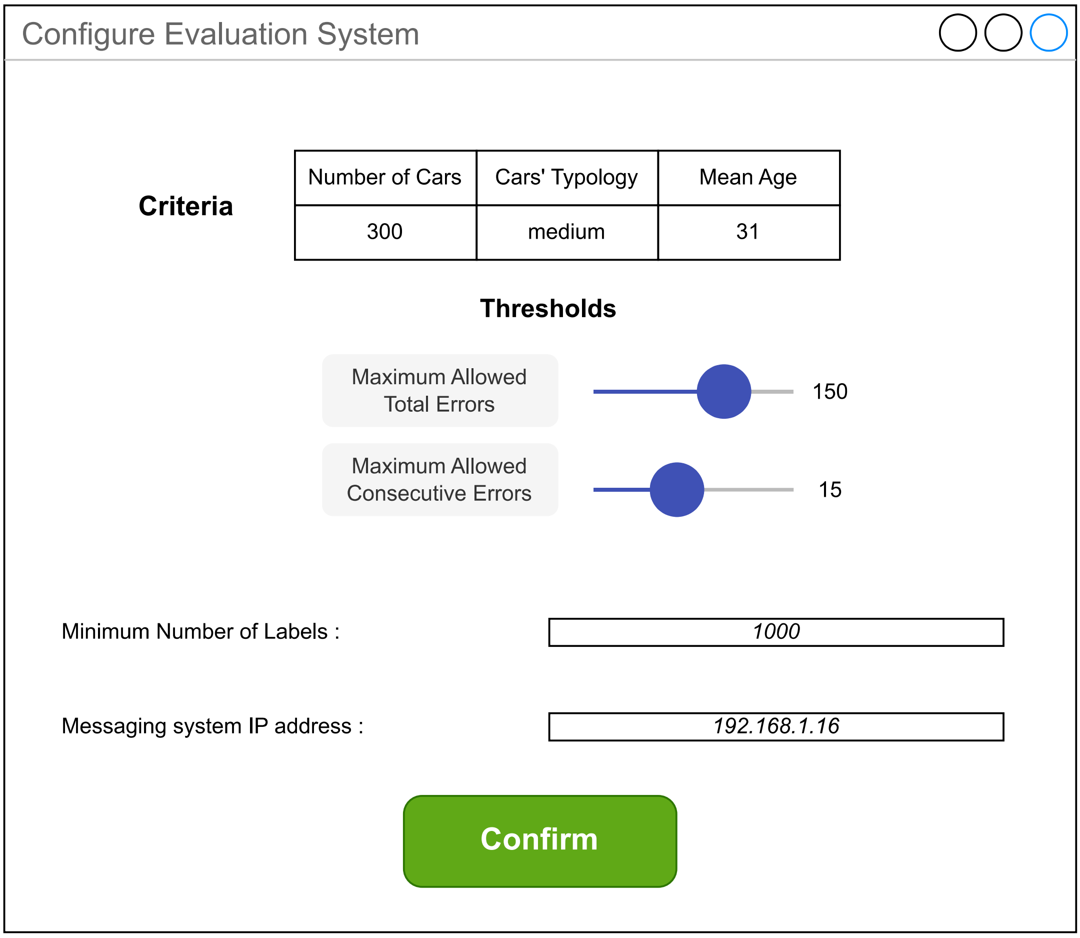
USER is a *Data Analyst*



STEP	COST CALCULATION	STEP COST
1 USER opens production system configuration page	1 x 1 x 1	1
2 SYSTEM shows production system configuration page		
3 USER checks the 3 criteria	3 x 4 x 1	12
4 USER sets the number of production sessions	1 x 4 x 1 x 3	12
5 USER sets the number of evaluation sessions	1 x 4 x 1 x 3	12
6 USER sets the IP address of Evaluation system	1 x 1 x 1	1
7 USER sets the IP address of Client side system	1 x 1 x 1	1
8 USER sets the IP address of Messaging system	1 x 1 x 1	1
9 SYSTEM enables the "CONFIRM" button		
10 USER presses confirm	1 x 1 x 1	1
11 USER closes production system configuration page	1 x 1 x 1	1
	HUMAN TASK COST	**42**

4.3.6 EVALUATION SYSTEM (NICCOLAI)

USER is a *Machine Learning Engineer*



STEP	COST CALCULATION	STEP COST
1 USER opens evaluation system configuration page	1 x 1 x 2.5	2.5
2 SYSTEM shows evaluation system configuration page		
3 USER checks the 3 criteria	3 x 4 x 2.5	30
4 USER sets the maximum allowed for total errors	1 x 4 x 2.5 x 3	30
5 USER sets the maximum allowed for consecutive errors	1 x 4 x 2.5 x 3	30
6 USER sets the minimum number of labels	1 x 4 x 2.5 x 3	30
7 USER sets the IP address of Messaging system	1 x 1 x 2.5	2.5
8 SYSTEM enables the "CONFIRM" button		
9 USER presses confirm	1 x 1 x 2.5	2.5
10 USER closes evaluation system configuration page	1 x 1 x 2.5	2.5
	HUMAN TASK COST	**130**

4.4 HUMAN TASK

4.4.1 LABEL RECORDS (BELLO)

USER is an *Automobile Damage Appraiser*

Label Records

THEN

Previous photo

NOW

Next photos

UUID: 1053472

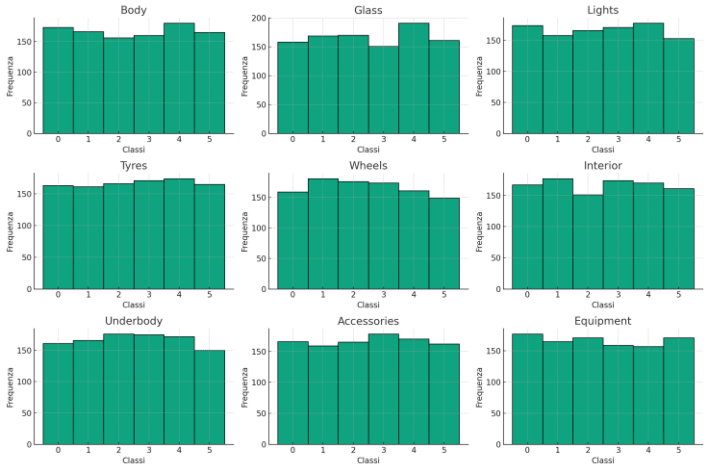
BODY	GLASS	LIGHTS	TYRES	WHEELS	INTERIOR	UNDERBODY	ACCESSORIES AND EQUIPMENT
No Damage	No Damage	No Damage	No Damage	No Damage	No Damage	No Damage	No Damage
Low	Low	Low	Low	Low	Low	Low	Low
Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium
High	High	High	High	High	High	High	High
Huge	Huge	Huge	Huge	Huge	Huge	Huge	Huge

STEP	COST CALCULATION	STEP COST
1 USER opens label records window	1 x 1 x 1.3	1.3
2 SYSTEM shows vehicles' photos and buttons to label the extent of damage		
3 FOR EACH photo	10	
3.1 USER compares begging and end photos	10 x 3 x 1.3	39
4 FOR EACH part of the auto	8	
4.1 USER clicks on one of the extent damage buttons	8 x 1 x 1.3	10.4
5 USER closes label records window	1 x 1 x 1.3	1.3
	HUMAN TASK COST	**52**

4.4.2 CHECK DATA BALANCING (FRATI)

USER is a *Data Analyst*

CHECK DATA BALANCING



	Body	Glass	Lights	Tyres	Wheels	Interior	Underbody	Accessories	Equipment
Class 0	173	158	174	163	159	167	161	166	177
Class 1	166	*169*	158	161	181	177	166	159	165
Class 2	156	*170*	166	166	176	151	176	165	171
Class 3	160	151	171	171	174	174	175	178	159
Class 4	180	191	178	174	161	170	172	170	157
Class 5	165	161	153	165	149	161	150	162	171
Mean	**166.7**	**166.7**	**166.7**	**166.7**	**166.7**	**166.7**	**166.7**	**166.7**	**166.7**
# exceeding classes	**2**	**3**	**3**	**0**	**3**	**2**	**2**	**1**	**2**

Accept balancing   Refuse balancing

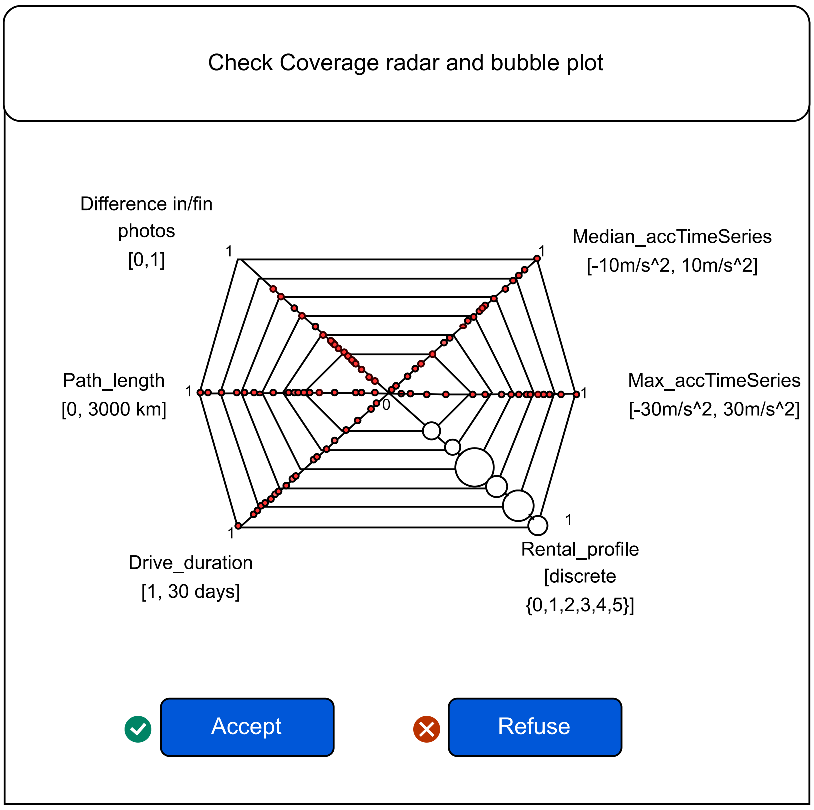
STEP	COST CALCULATION	STEP COST
1 USER open check data balancing window	1 x 1 x 1	1
2 SYSTEM shows the histogram of attributes with scaled distribution and the table with balancing scores		
3 FOR each feature	9 iterations	
3.1 USER checks the distribution	9 x 4 x 1	36
3.2 USER decides if number of exceeding class is too high	9 x 4 x 1	36
4.1 IF no criticities are detected	0.2	
4.1.1 USER selects "ACCEPT BALANCING"	0.2 x 1 x 1	0.2
4.2 ELSE	0.8	
4.2.1 USER selects "REFUSE BALANCING"	0.8 x 1 x 1	0.8
5 USER closes check data balancing window	1 x 1 x 1	1
	HUMAN TASK COST	**75**

The data analyst checks the distribution for each feature. Some histograms and a table that shows the distribution of each output class for each output feature are shown. The data analyst compares these with the expected ones and decides if the balancing is acceptable or not.

These decisions require a big cognitive effort, related also to previous knowledge and experience.

4.4.3 CHECK COVERAGE (FRATI)

USER is a *Data Analyst*

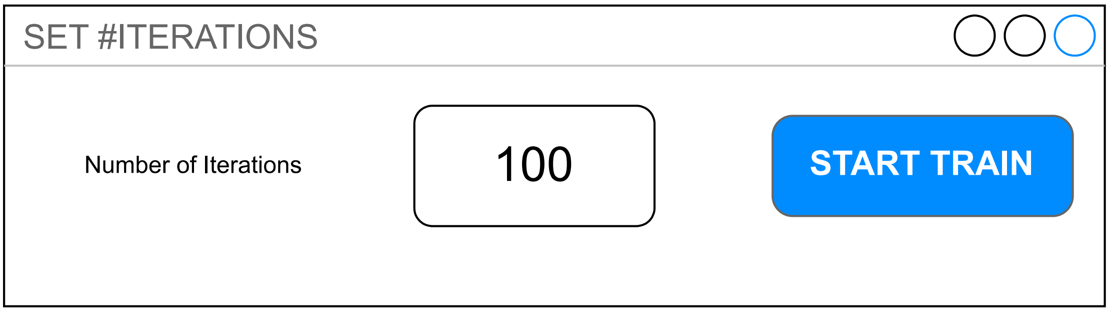


STEP	COST CALCULATION	STEP COST
1 USER open check data coverage window	1 x 1 x 1	1
2 SYSTEM shows the radar and bubble plot with scaled distribution of the observations of the dataset		
3 FOR each feature	6 iterations	
3.1 USER checks the feature's distribution	6 x 4 x 1	24
3.2 USER thinks about expected distribution	6 x 4 x 1	24
3.3 USER compare expected and real distribution	6 x 4 x 1	24
4 IF coverage is not satisfied	0.67 x 1 x 1	
4.1 USER thinks about missing data	0.67 x 4 x 1 x 1	2.67
4.2 USER select "REFUSE"	0.67 x 1 x 1	0.67
3.2 ELSE	0.33	
3.2.1 USER selects "ACCEPT"	0.33 x 1 x 1	0.33
4 USER closes check coverage window	1 x 1 x 1	1
	HUMAN TASK COST	**77.67**

The data analyst checks the radar plot to decide if the input coverage is sufficient or not. Similarly to what happens in check data balancing, there are different phases with high cognitive effort: to check the real distribution, to think about the expected one and to compare them. Basing on experience data analyst can evaluate the quality of coverage guaranteed.

4.4.4 SET #ITERATIONS (BELLO)

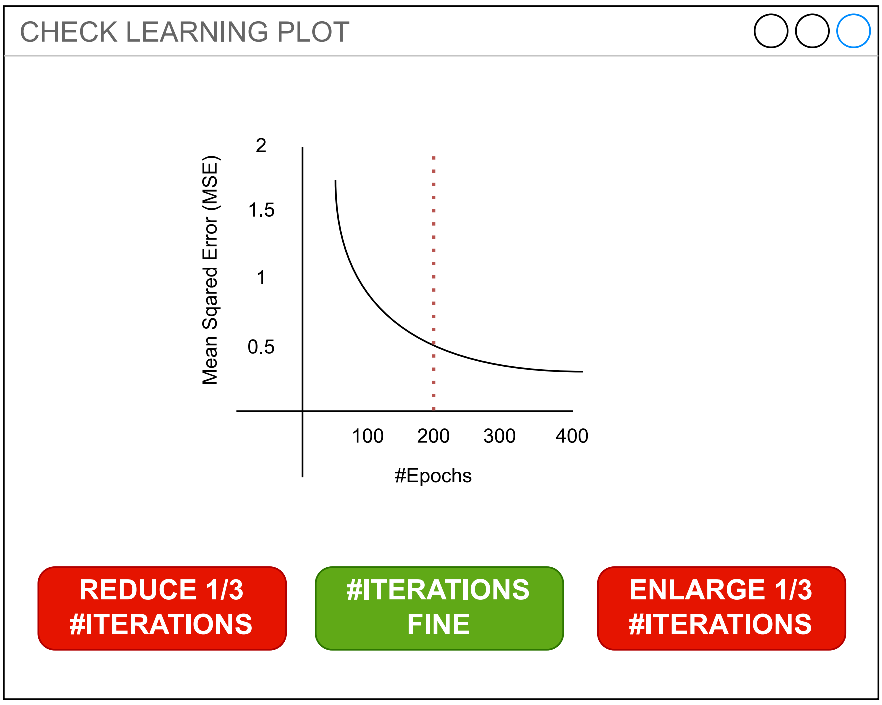
USER is a *Machine Learning Engineer*



STEP	COST CALCULATION	STEP COST
1 USER opens set #iterations window	1 x 1 x 2.5	2.5
2 SYSTEM shows the number of iterations and "START TRAIN" button		
3 USER may change the number of iterations	0.25 x 4 x 2.5	2.5
4 USER clicks on the "START TRAIN" button	1 x 1 x 2.5	2.5
5 USER closes the set #iterations windows	1 x 1 x 2.5	2.5
	HUMAN TASK COST	**10**

4.4.5 CHECK LEARNING PLOT (BELLO)

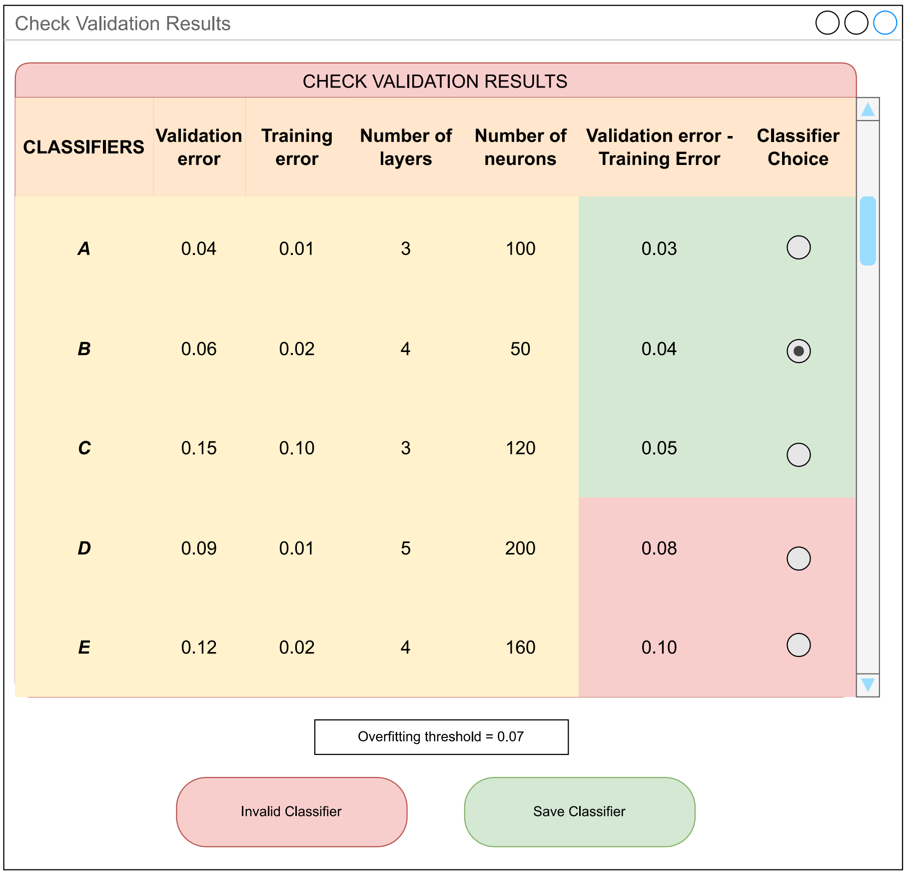
USER is a *Machine Learning Engineer*



STEP	COST CALCULATION	STEP COST
1 USER opens check learning plot window	1 x 1 x 2.5	2.5
2 SYSTEM shows the learning plot and #iterations "FINE" "ENLARGE" "REDUCE" buttons		
3 USER checks the learning plot	1 x 3 x 2.5	7.5
3.1 IF the learning plot is good	.2	
3.1.1 USER clicks on the "#ITERATIONS FINE" button	.2 x 1 x 2.5	0.5
3.2 ELSE IF the loss is flat for at least the half	.4	
3.2.1 USER clicks on the "#ITERATIONS ENLARGE" button	.4 x 1 x 2.5	1
3.3 ELSE IF the loss is not flat at the end of the iterations	.4	
3.3.1 USER clicks on the "#ITERATIONS REDUCE" button	.4 x 1 x 2.5	1
4 USER closes the chek learning plot windows	1 x 1 x 2.5	2.5
	HUMAN TASK COST	**15**

4.4.6 CHECK VALIDATION RESULTS (CAPECCHI)

USER is a Machine Learning Engineer

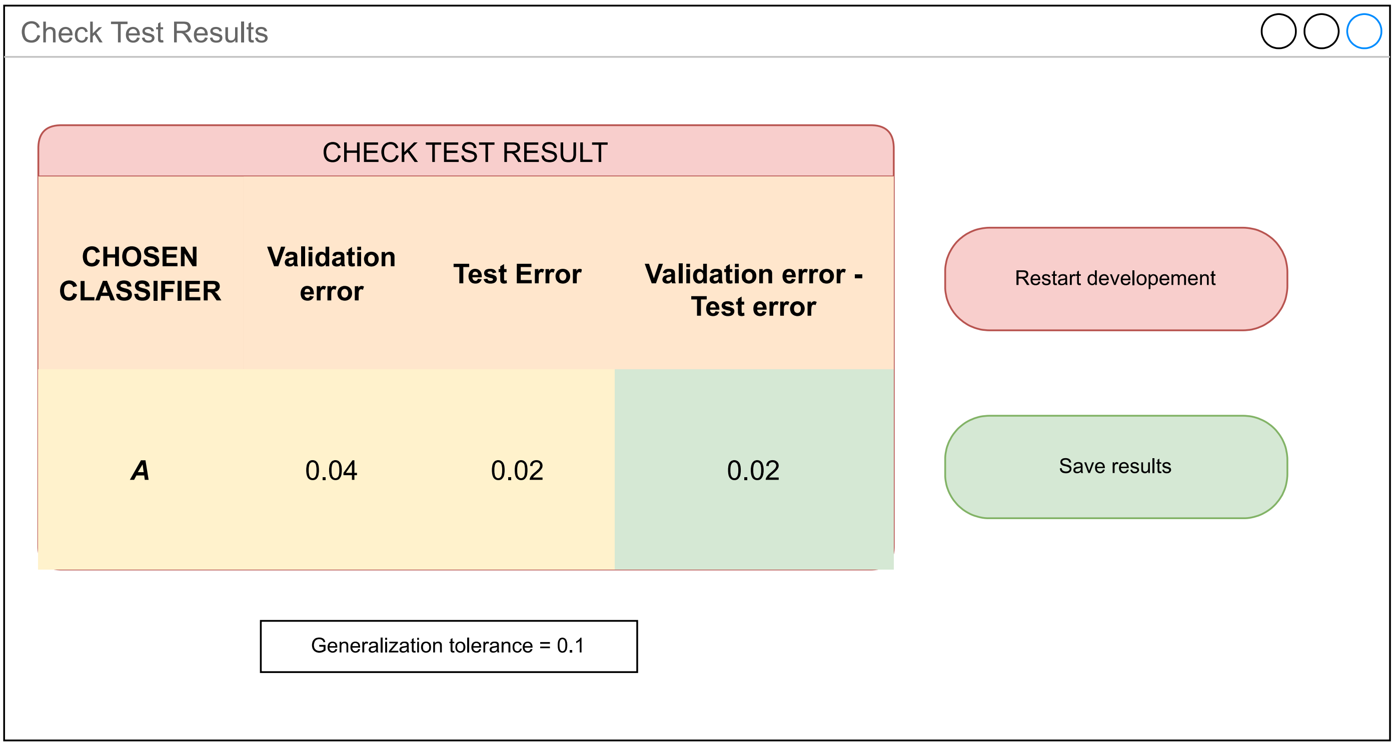


STEP	COST CALCULATION	STEP COST
1 USER opens validation results window	1 x 1 x 2.5	2.5
2 SYSTEM shows the validation results table		
3 USER checks the classifiers' errors and the threshold	1 x 3 x 2.5	7.5
3.1 IF no valid classifier is present	.05	
3.1.1 USER selects "INVALID CLASSIFIER"	1 x 1 x 2.5	2.5
3.2 ELSE	.95	
3.2.1 USER confronts classifiers with comparable Validation error - Training error	.95 x 5 x 3 x 2.5	35.625
3.2.1.1 IF there aren't comparable results	.1	
3.2.1.1.1 USER selects first classifier proposed	.95 x .1 x 1 x 2.5	0.2375
3.2.1.2 ELSE	.9	
3.2.1.2.1 USER computes the product between number of layers and number of nodes for each classifier	.95 x .9 x 3 x 2.5	6.4125
3.2.1.2.2 USER selects the classifier with the lower total number of nodes	.95 x .9 x 1 x 2.5	2.1375
3.3 SYSTEM enables "SAVE CLASSIFIER" button		
3.4 USER presses "SAVE CLASSIFIER"	.95 x 3 x 2.5	7.125
4 USER closes data balancing form	1 x 1 x 2.5	2.5
	HUMAN TASKS COST	**67,5375**

Operation 3.2.1.1 has a 10% probability of happening; the reason is that we imagine that for most of the cases (90%), since the data are already checked in the previous systems and the expertise of the users in play is high, the results will be comparable between the obtained classifiers. In green we have a multiplication because operation 3.2.1 is done for each of the five classifiers.

4.4.7 CHECK TEST RESULTS (CAPECCHI)

USER is a *Machine Learning Engineer*

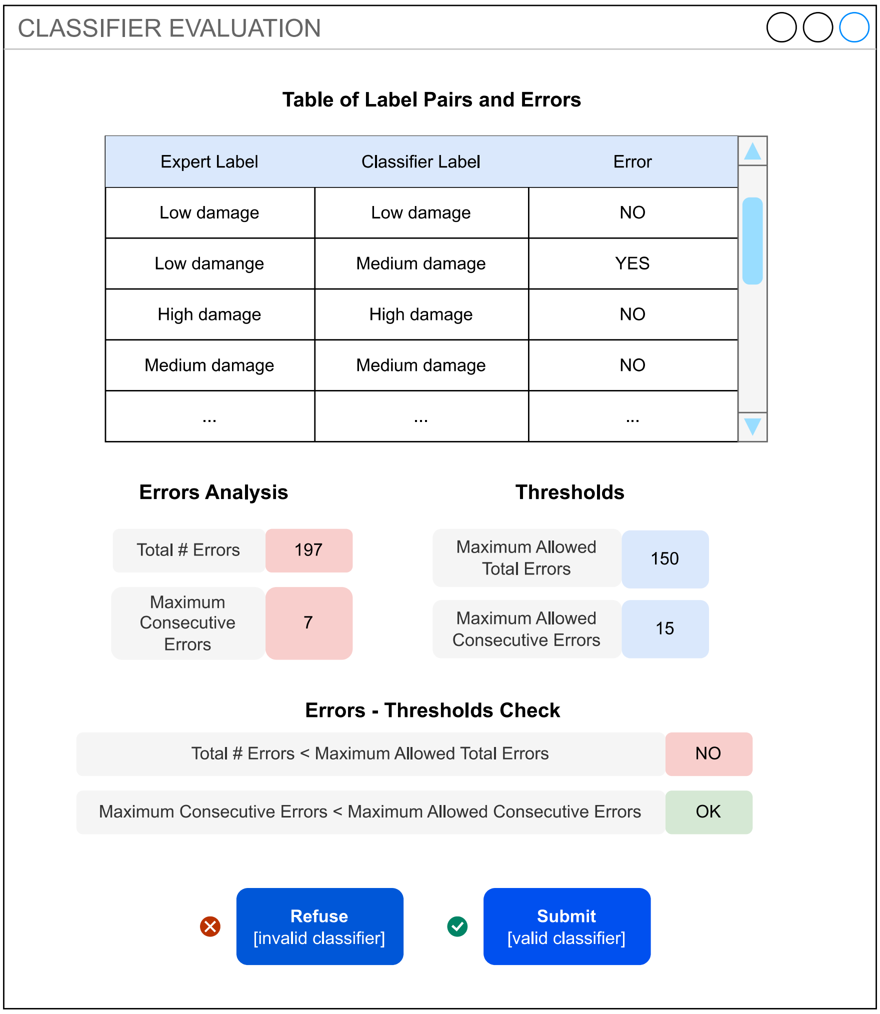


STEP	COST CALCULATION	STEP COST
1 USER opens the test results window	1 x 1 x 2.5	2.5
2 SYSTEM shows the test results table		
3 USER confronts the difference between validation error and test error with the threshold	1 x 3 x 2.5	7.5
3.1 IF value is lower than the threshold	.99	
3.1.1 USER presses "SAVE RESULTS"	.99 x 1 x 2.5	2.475
3.2 ELSE if value is greater or equal than the threshold	.01	
3.2.1 USER selects "RESTART DEVELOPEMENT"	.01 x 1 x 2.5	0.025
4 USER closes test results form	1 x 1 x 2.5	2.5
	HUMAN TASKS COST	**15**

Costs are generally low because in this phase the Machine Learning Engineer only has to check if the obtained result is fine, with the information already provided by the interface.

4.4.8 EVALUATE CLASSIFIER (NICCOLAI)

USER is a *Machine Learning Engineer*



STEP	COST CALCULATION	STEP COST
1 USER opens classifier evaluation window	1 x 1 x 2.5	2.5
2 SYSTEM shows table of labels and errors, thresholds conditions		
3 USER checks table and values	1 x 2 x 2.5	5
3.1 IF total # errors < maximum allowed total errors AND maximum consecutive errors < maximum allowed consecutive errors	0.86	
3.1.1 SYSTEM enables the submit button		
3.1.2 USER selects SUBMIT	0.86 x 1 x 2.5	2.15
3.2 ELSE	0.14	
3.2.1 SYSTEM disables the submit button		
3.2.2 USER selects REFUSE	0.14 x 1 x 2.5	0.35
4 USER closes classifier evaluation window	1 x 1 x 2.5	2.5
	HUMAN TASK COST	**12.5**

4.5 DATA MODEL

4.5.1 PREPARE SESSION (CAPECCHI)

4.5.1.1 ingestion system

The record is formed by:

1. blackbox => UUID, drivingDynamics(gpsTimeSeries, accelerationTimeSeries)
2. gateCameras => UUID, photos(front, back, dx, sx, internal) beginning/end
3. Car Rental Management System => UUID, rentalProfile(vehicleCategory, driverAge).

Here we represent it in a compact way, decomposing the blackbox, gate cameras, and the client information in their components, and by linking them through the UUID.

INGESTION SYSTEM CONFIGURATION PARAMETERS
minimum # of records to start with :integer max # of missing samples :integer current # of records :integer evaluation system ip address :string preparation system ip address :string evaluation phase :boolean development phase :boolean

RECORD
UUID :integer gpsTimeSeries :list<pair<DateTime,coordinate> accelerationTimeSeries :list<pair<DateTime,float>> photos :list<BufferedImages> veichleCategory :integer driverAge :integer label :integer {optional}

RAW SESSION
UUID :integer gpsTimeSeries :list<pair<DateTime,coordinate> accelerationTimeSeries :list<pair<DateTime,float>> photos :list<BufferedImages> veichleCategory :integer driverAge :integer label :integer {optional}

LABEL EXPERT
body :integer glass :integer lights :integer tyres :integer wheels :integer interior :integer underbody :integer accessories and equipment :integer UUID :integer

4.5.1.2 preparation system

Here, we represent into the “Preparation System Parameters” data object the possible min/max values for each value that can be (e.g. for interference problems) out of bound. The raw session, received from the segregation system, is then transformed into the prepared session, where the features are extracted.

PREPARATION SYSTEM PARAMETERS
MAX gpsTimeSeries :coordinate
MIN gpsTimeSeries :coordinate
MAX accelerationTimeSeries :float
MIN accelerationTimeSeries :float
MAX driverAge :integer
MIN driverAge :integer
MAX photoBluriness :float
MIN photoBlurriness :float
features to extract :boolean array
production system ip address :string
segregation system ip address :string
development phase :boolean

RAW SESSION INPUT
UUID :integer
gpsTimeSeries :list<pair<DateTime,coordinate>
accelerationTimeSeries :list<pair<DateTime,float>>
photos :list<BufferedImages>
veichleCategory :integer
driverAge :integer
label :integer {optional}

PREPARED SESSION
max acceleration :float
median acceleration :float
driving duration :float
path length :float
difference initial final photos :float
rental profile :integer
session id :integer
expert label :integer[] {optional}

4.5.2 GENERATE LEARNING SET (FRATI)

PREPARED SESSION
max acceleration :float
median acceleration :float
driving duration :float
path length :float
difference initial final photos :float
rental profile :integer
session id :integer
expert label :integer[]
flag sufficient session :boolean

LEARNING SET
max acceleration :float
median acceleration :float
driving duration :float
path length :float
difference initial final photos :float
rental profile :integer
session id :integer
expert label :integer[]

SEGREGATION PARAMETERS
min number rows :integer
percentual deviation from mean :float
messaging system ip :string
development system ip :string
preparation system ip :string

CLASSES BALANCEMENT REPORT
session Id :integer
message :string
umbalanced classes :integer[]

COVERAGE REPORT
session Id :integer
message :string
umbalanced features :string[]

The learning set is a set of prepared sessions. The expert label is made by a list of integers that identify the damages of the car in the checked parts.

A flag to detect if the session is sufficient or not is present.

The classes balance report contains a message field where the data analyst can insert the problems found in the set. A list of integers is added to identify the unbalanced ones.

The coverage report contains a message field too, where the data analyst can insert which are the problems in the coverage of the input classes. An array of string is added to allow the data analyst to write which are the unbalanced features.

4.5.3 DEVELOP CLASSIFIER (BELLO)

VALIDATION PARAMETERS
min layers :integer step layers :integer max layers :integer min neurons :integer step neurons :integer max neurons :integer initial #iterations :integer overfitting threshold :float generalization tolerance :float actual layers :integer actual neurons :integer
set average hyperparameters (min layers, max layers, min neurons, max neurons) :integer[2] set hyperparameters (actual layers, step layers, actual neurons, step neurons) :integer[2]

FLAG GATEWAY
ongoing validation :boolean iterations fine :boolean valid classifier :boolean

MESSAGGE CONFIGURATION
test passed :boolean final neurons :integer final layers :integer

IP CONFIGURATION
messaging system ip :String segregation system ip :String

LEARNING SET
max acceleration :float median acceleration :float driving duration :float path length :float difference initial final photos :float rental profile :integer session id :integer expert label :integer[]

4.5.4 CLASSIFY SESSION (NICCOLAI)

PREPARED SESSION
max acceleration :float median acceleration :float driving duration :float path length :float difference initial final photos :float rental profile :integer session id :integer expert label :integer[]

PRODUCTION PARAMETERS
number of production sessions :integer number of evaluation sessions :integer evaluation system ip address :string client side system ip address :string messaging system address :string evaluation phase flag :boolean

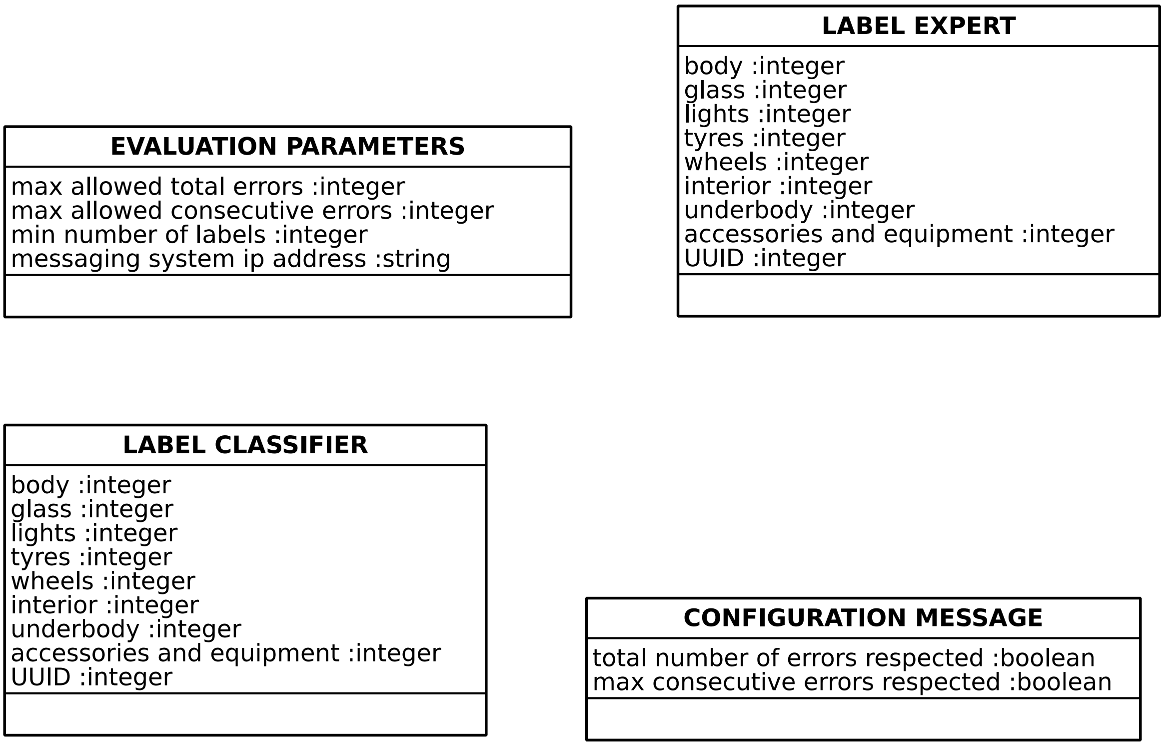
LABEL
body :integer glass :integer lights :integer tyres :integer wheels :integer interior :integer underbody :integer accessories and equipment :integer UUID :integer

CONFIGURATION MESSAGE
number of neurons :integer number of iterations :integer number of layers :integer

The evaluation phase flag in Production Parameters was introduced to guarantee the presence of an essential information required to undertake the corresponding gateway.

The integer array in Prepared Session corresponds to the attributes listed in the Label data object.

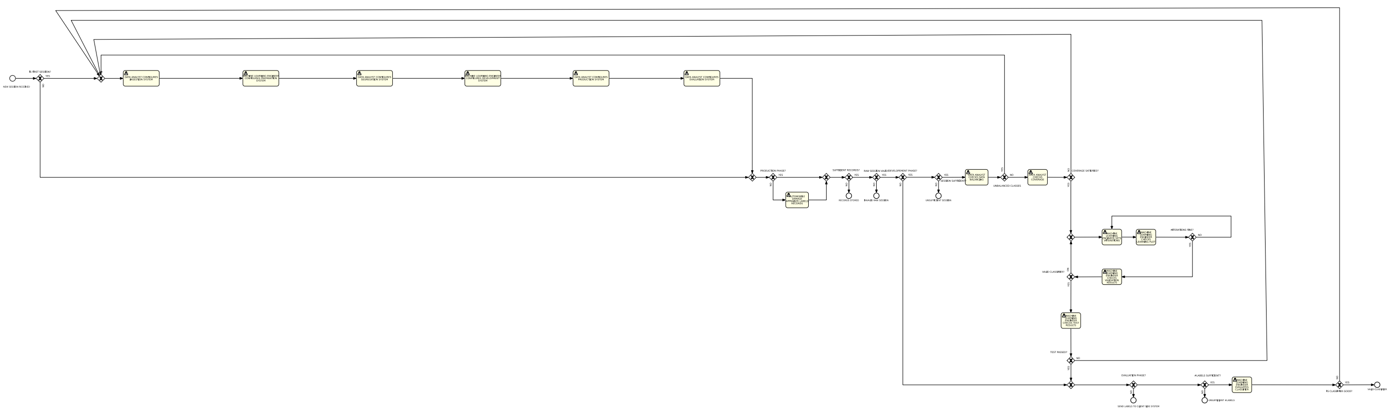
4.5.5 EVALUATE CLASSIFIER PERFORMANCE (NICCOLAI)



5 AS-IS

5.1 COLLAPSED LAYOUT (BELLO, CAPECCHI, FRATI, NICCOLAI)

The collapsed layout is a result of an aggregation of the precedents BPMN diagrams but considering only the human tasks. As we can see there are four different times where, for specific reasons, we go back to configure each system. The start event is the receiving of a new session, and there are multiple end events.



Assumptions to buildup Gateway probabilities:

GATEWAY	PROBABILITIES (%)		REASON
	TRUE CASE	FALSE CASE	
Is first session?	0.02	99.98	We have 500 sessions of development, 5000 of production and 50 of evaluation. In total 5000+500+50 = 5550, so the probability of being in the first one is 1/5550 that is approximately 0.02
Production phase?	90.1	9.9	5000 phases of production on 5550 -> 5000/5550=90.1
Sufficient records?	10	90	In most of the cases, it will be necessary to wait for the arrival of all the photos, since these are the heaviest object to transfer
Raw session valid?	99	1	We assumed most of the times it will be valid, except for the cases where e.g. we have a broken GPS, accelerometer, or camera, that will invalidate it
Development phase?	9	91	We considered 500 evaluation sessions on a total of 5550 sessions
Session sufficient?	99	1	We assumed that in the great majority of the cases the session is sufficient.
Unbalanced classes?	80	20	We assumed that every 5 occurrences only 1 has balanced classes
Coverage satisfied?	33	67	In 3 iterations only 1 has a sufficient coverage of the input features
# iterations fine?	80	20	We assumed that every 5 iterations only 1 time we have a fine number
Valid classifier?	95	5	Most of the times the results will be acceptable
Test passed?	99	1	If the classifier is valid (previous gateway), it will most likely pass the test
Evaluation phase?	0.9	99.1	We considered 50 evaluation sessions on a total of 5550 sessions
# labels sufficient?	90	10	We assumed that most of the times the # of labels will be sufficient
Is classifier good?	86	14	Considering 7 iterations, for 6 of these the evaluation is fine

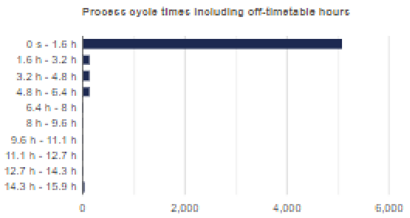
5.2 AS-IS BIMP SIMULATION (BELLO, NICCOLAI)

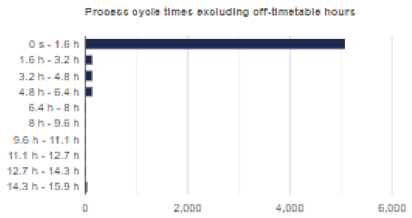
5.2.1 RESULTS

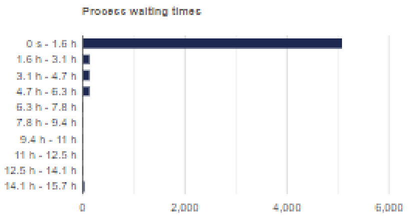
General information

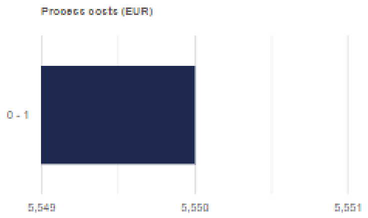
Completed process instances	5550
Total cost	0 EUR
Total simulation time	16.5 hours

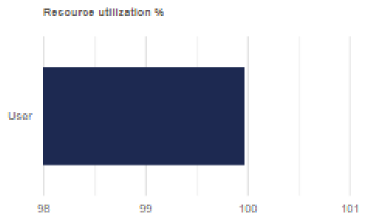
Charts











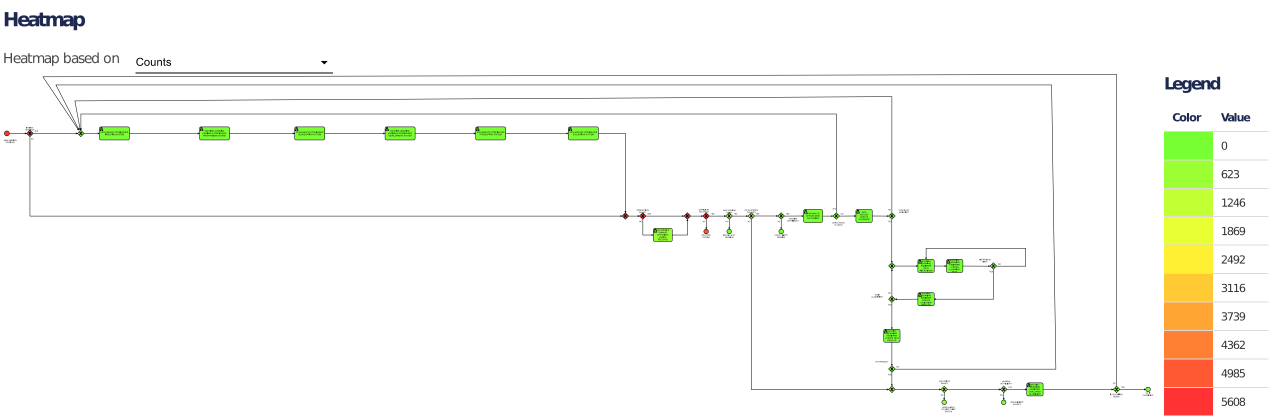
Scenario Statistics

	Minimum	Maximum	Average
Process instance cycle times including off-timetable hours	0 seconds	15.9 hours	26.9 minutes
Process instance cycle times excluding off-timetable hours	0 seconds	15.9 hours	26.9 minutes
Process instance costs	0 EUR	0 EUR	0 EUR

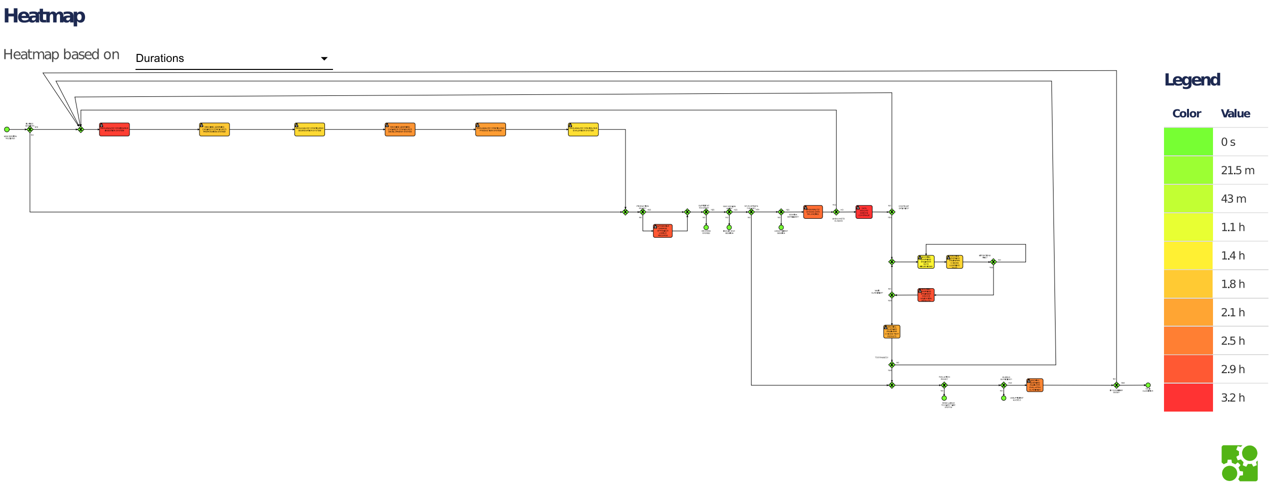
Activity Durations, Costs, Waiting times, Deviations from Thresholds

Name		Waiting time			Duration			Duration over threshold			Cost			Cost over threshold		
	Count	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
AUTOMOBILE DAMAGE APPRAISER LABELS RECORDS	562	0 s	3.1 h	6.1 h	43.2 s	3.1 h	6.2 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CHECKS COVERAGE	9	1.4 h	3.9 h	6.1 h	1.4 h	4 h	6.1 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CHECKS DATA BALANCING	48	20.1 m	2.9 h	6.2 h	21.4 m	2.9 h	6.2 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CONFIGURES EVALUATION SYSTEM	44	16.4 m	1.3 h	2.5 h	18.6 m	1.3 h	2.6 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CONFIGURES INGESTION SYSTEM	44	49.4 m	3.3 h	6.1 h	49.9 m	3.3 h	6.2 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CONFIGURES PRODUCTION SYSTEM	44	1 h	1.8 h	2.6 h	1.1 h	1.8 h	2.6 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CONFIGURES SEGREGATION SYSTEM	44	50.1 m	1.3 h	2.5 h	50.8 m	1.3 h	2.5 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CHECKS LEARNING PLOT	5	1.2 h	1.3 h	1.4 h	1.2 h	1.3 h	1.4 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CHECKS TEST RESULTS	5	1.4 h	1.8 h	2.6 h	1.4 h	1.8 h	2.6 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CHECKS VALIDATION RESULTS	5	1.5 h	2.2 h	2.5 h	1.5 h	2.2 h	2.6 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CONFIGURES DEVELOPMENT SYSTEM	44	1.1 h	1.8 h	2.5 h	1.2 h	1.9 h	2.6 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CONFIGURES PREPARATION SYSTEM	44	49.4 m	1.6 h	5.5 h	51.6 m	1.6 h	5.5 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER EVALUATES CLASSIFIER	10	50.8 m	3.1 h	5.9 h	51 m	3.1 h	5.9 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER SETS #ITERATIONS	5	50.3 m	1.1 h	1.2 h	50.5 m	1.1 h	1.2 h	0 s	0 s	0 s	0	0	0	0	0	0

5.2.2 HEATMAP BASED ON COUNTS



5.2.3 HEATMAP BASED ON DURATIONS



Duration has been set to reflect the costs of the activities, in fact as expected the tasks with a red colour are the most expensive ones.

A particular discussion must be made for the first activity found after the start event. It is *Configure Ingestion System* and looking at the picture it could seem as it has a really high cost. In truth, the problem is that there is a queue since the tokens produced in the start cannot be disposed of in time, and this leads to an increase in the total duration.

We carried out a test by decreasing the token frequency rate and noticed the problem disappearing. This certifies that the red colour associated with the task is not due to its cost, but to the problem previously exposed.

6 TO-BE

6.1 Improvements (CAPECCHI, FRATI)

We made some improvements at three different levels:

1. *Handoff level*: two gateways were dropped (*Session sufficient?* and *Unbalanced classes?*). We imagined having a database with information of similar networks in the same category. This allows us to take sessions directly there instead of collecting new ones. This way we can also skip a reconfiguration.
2. *Service level*: a human task was removed (*Set # iterations*), because it could easily be replaced by an automatic script that takes this information by a cluster of similar companies.
3. *Task level*: improvement of a Data Analyst's task. In Check Data Balancing the cognitive effort of the actor has been reduced from 4 (analyse) to 2 (understand) for the check of the exceeding classes. In fact, is possible to have some thresholds set considering previous experiences of similar companies.

This reduced the total cost of this human task from **75** to **57.** In red, we can see that before the Data Analyst has to *decide* if the number of exceeding classes was too high, based on his previous knowledge. Now, he only has to check since this information is provided by the user interface.

STEP	COST CALCULATION	STEP COST
1 USER open check data balancing window	1 x 1 x 1	1
2 SYSTEM shows the histogram of attributes with scaled distribution and the table with balancing scores		
3 FOR each feature	9 iterations	
3.1 USER checks the distribution	9 x 4 x 1	36
3.2 USER checks if number of exceeding class is too high	9 x 2 x 1	18
4.1 IF no criticities are detected	0.2	
4.1.1 USER selects "ACCEPT BALANCING"	0.2 x 1 x 1	0.2
4.2 ELSE	0.8	
4.2.1 USER selects "REFUSE BALANCING"	0.8 x 1 x 1	0.8
5 USER closes check data balancing window	1 x 1 x 1	1
	HUMAN TASK COST	57

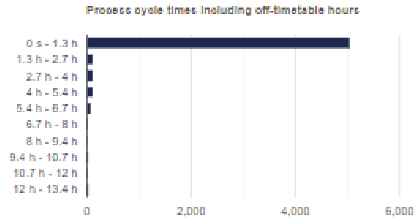
6.2 TO-BE BIMP SIMULATION (CAPECCHI, FRATI)

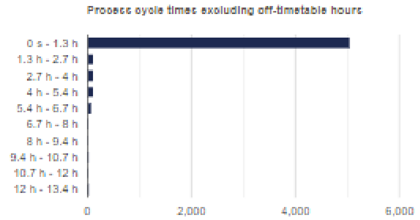
6.2.1 RESULTS

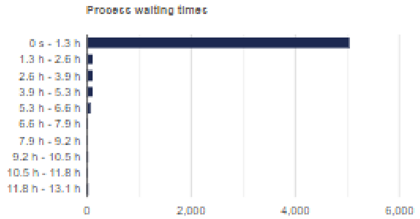
General information

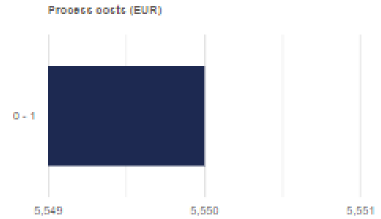
Completed process instances	5550
Total cost	0 EUR
Total simulation time	13.8 hours

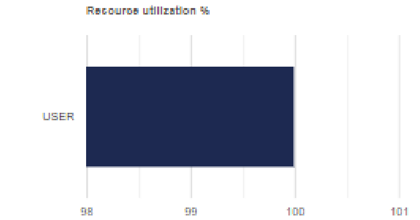
Charts











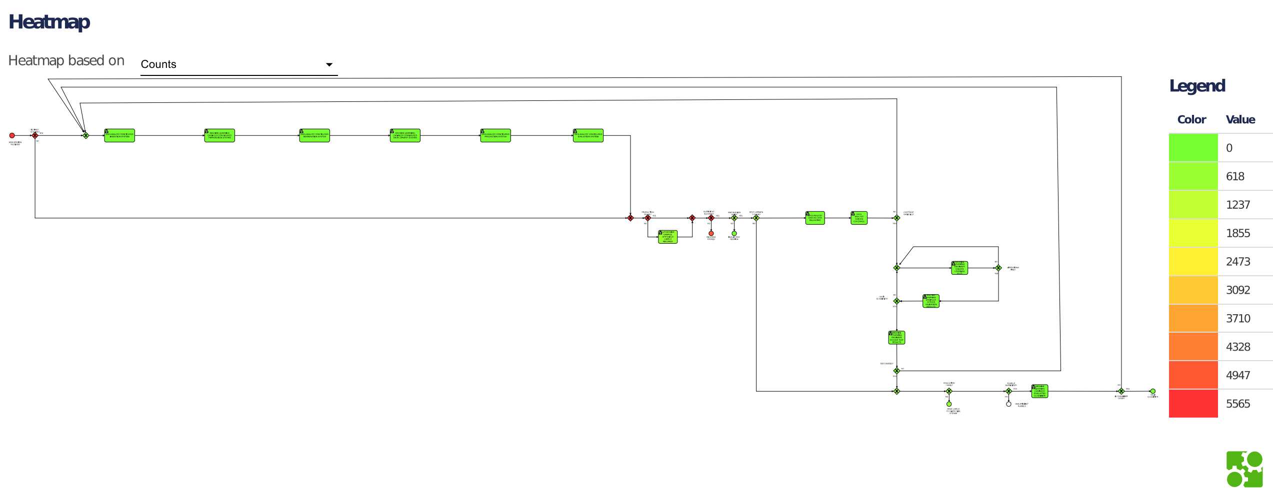
Scenario Statistics

	Minimum	Maximum	Average
Process instance cycle times including off-timetable hours	0 seconds	13.4 hours	25.4 minutes
Process instance cycle times excluding off-timetable hours	0 seconds	13.4 hours	25.4 minutes
Process instance costs	0 EUR	0 EUR	0 EUR

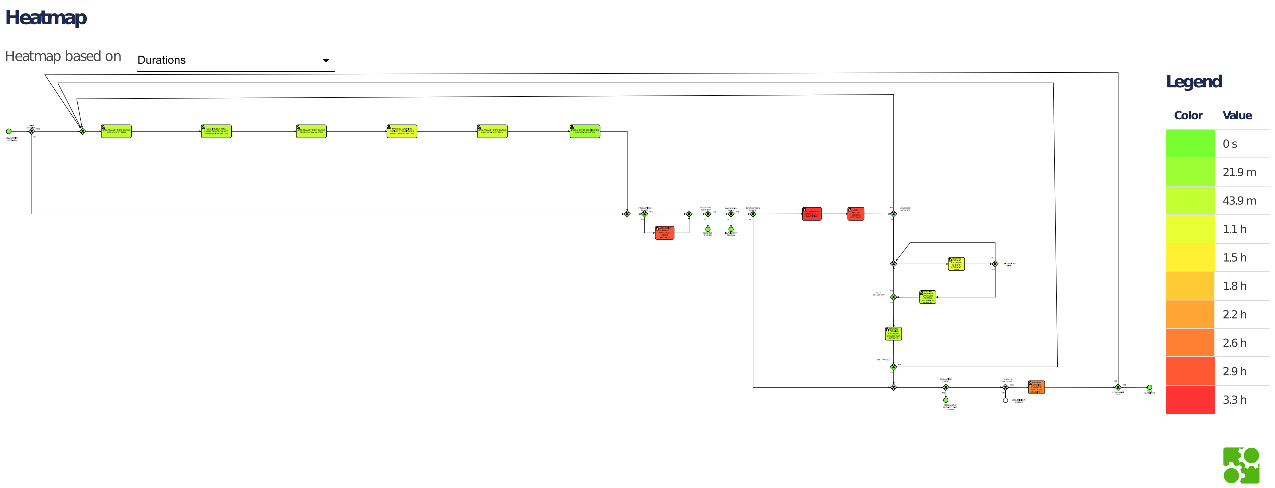
Activity Durations, Costs, Waiting times, Deviations from Thresholds

Name		Waiting time			Duration			Duration over threshold			Cost			Cost over threshold		
	Count	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
AUTOMOBILE DAMAGE APPRAISER LABELS RECORDS	572	0 s	3.1 h	6.2 h	41.1 s	3.1 h	6.2 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CHECKS COVERAGE	48	36 m	3.5 h	6.1 h	37.3 m	3.6 h	6.1 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CHECKS DATA BALANCING	48	9.4 m	3 h	6.2 h	10.4 m	3 h	6.2 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CONFIGURES EVALUATION SYSTEM	23	22.1 m	39 m	1.4 h	24.2 m	41.1 m	1.5 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CONFIGURES INGESTION SYSTEM	23	33.4 m	1.8 h	5.9 h	33.9 m	1.8 h	5.9 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CONFIGURES PRODUCTION SYSTEM	23	22.7 m	58.3 m	1.4 h	23.4 m	59 m	1.4 h	0 s	0 s	0 s	0	0	0	0	0	0
DATA ANALYST CONFIGURES SEGREGATION SYSTEM	23	36 m	57 m	1.2 h	36.7 m	57.7 m	1.2 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CHECKS LEARNING PLOT	33	34.4 m	1.1 h	4.7 h	34.6 m	1.1 h	4.7 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CHECKS TEST RESULTS	27	31.1 m	58.5 m	1.4 h	31.3 m	58.8 m	1.4 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CHECKS VALIDATION RESULTS	29	35.3 m	55.2 m	1.4 h	36.5 m	56.4 m	1.4 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CONFIGURES DEVELOPMENT SYSTEM	23	41 m	1 h	1.4 h	46 m	1.1 h	1.5 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER CONFIGURES PREPARATION SYSTEM	23	33.3 m	1.1 h	4.8 h	35.6 m	1.2 h	4.8 h	0 s	0 s	0 s	0	0	0	0	0	0
MACHINE LEARNING ENGINEER EVALUATES CLASSIFIER	4	1.6 h	3.8 h	5.3 h	1.6 h	3.8 h	5.3 h	0 s	0 s	0 s	0	0	0	0	0	0

6.2.2 HEATMAP BASED ON COUNTS



6.2.3 HEATMAP BASED ON DURATIONS



6.3 AS-IS & TO-BE MODEL COMPARISON (BELLO, CAPECCHI, FRATI, NICCOLAI)

Looking at the heatmaps is possible to note the effects of the improvements:

- Less returns to the default gateway
- Less costs in *Check Data Balancing* task
- General costs reduction in develop classifier's tasks

Is interesting to notice that the problem with *Configure Ingestion System* is not present in the to-be modeling simulation. This is because the great majority of tokens that return to the configuration phase in the as-is diagram come from the deleted gateway ("*Balanced classes?*") that has a percentage of *NO* equal to 80%. When the *NO* branch is selected, the tokens return to the configuration phase, causing a queue. This is avoided in the to-be diagram, thanks to the lower number of returning back tokens.

7 NORMATIVE PROCESS with single start and single end (Bello Niccolai)

To carry on with this phase, the BIMP simulation was modified in the following manner:

- Added a unique "START" and "END" event
- Default 1 euro cost and 1 sec duration to each task
- 10 lanes
- 50% to each gateway
- 100 input tokens

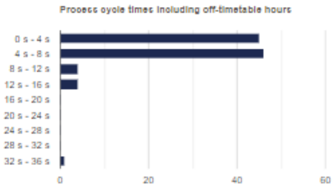
7.1 BIMP SIMULATION

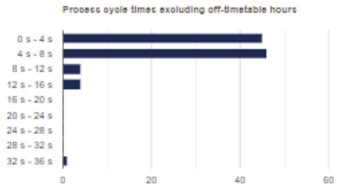
7.1.1 RESULTS

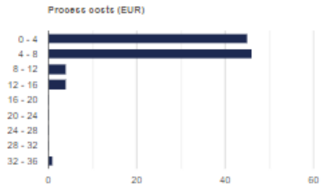
General information

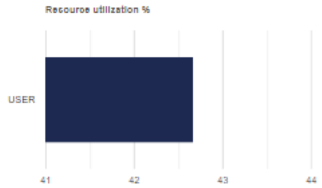
Completed process instances 100	
Total cost 448 EUR	
Total simulation time 1.8 minutes	

Charts









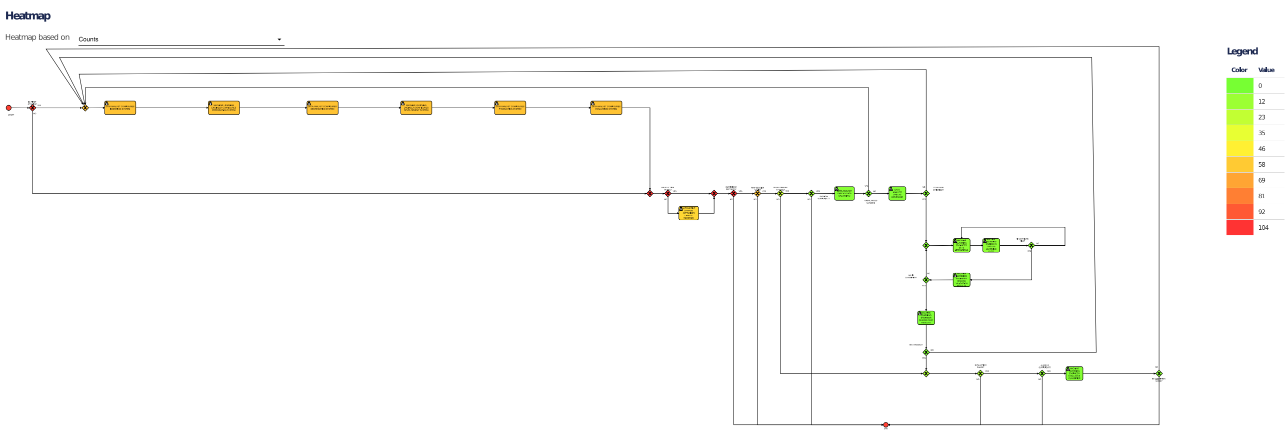
Scenario Statistics

	Minimum	Maximum	Average
Process instance cycle times including off-timetable hours	0 seconds	34 seconds	4.5 seconds
Process instance cycle times excluding off-timetable hours	0 seconds	34 seconds	4.5 seconds
Process instance costs	0 EUR	34 EUR	4.5 EUR

Activity Durations, Costs, Waiting times, Deviations from Thresholds

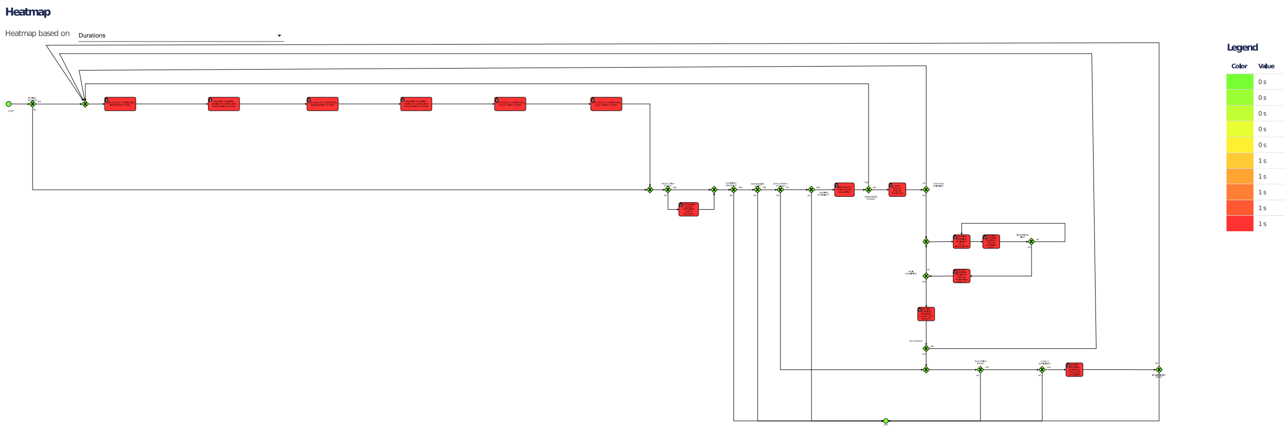
Name		Waiting time			Duration			Duration over threshold			Cost			Cost over threshold		
	Count	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
AUTOMOBILE DAMAGE APPRAISER LABELS RECORDS	55	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
DATA ANALYST CHECKS COVERAGE	5	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
DATA ANALYST CHECKS DATA BALANCING	6	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
DATA ANALYST CONFIGURES EVALUATION SYSTEM	58	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
DATA ANALYST CONFIGURES INGESTION SYSTEM	58	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
DATA ANALYST CONFIGURES PRODUCTION SYSTEM	58	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
DATA ANALYST CONFIGURES SEGREGATION SYSTEM	58	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
MACHINE LEARNING ENGINEER CHECKS LEARNING PLOT	12	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
MACHINE LEARNING ENGINEER CHECKS TEST RESULTS	2	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
MACHINE LEARNING ENGINEER CHECKS VALIDATION RESULTS	4	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
MACHINE LEARNING ENGINEER CONFIGURES DEVELOPMENT SYSTEM	58	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
MACHINE LEARNING ENGINEER CONFIGURES PREPARATION SYSTEM	58	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
MACHINE LEARNING ENGINEER EVALUATES CLASSIFIER	4	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0
MACHINE LEARNING ENGINEER SETS #ITERATIONS	12	0 s	0 s	0 s	1 s	1 s	1 s	0 s	0 s	0 s	1	1	1	0	0	0

7.1.2 HEATMAP BASED ON COUNTS



To correctly interpret the heatmap above, is important to remember that all the gateway's probabilities are set to 50% in this phase. This fact causes this effect to be created whereby the closer an activity is to the start, the greater the probability that a token passes through it. The concept of proximity can be expressed in terms of gateways crossed to reach the task itself from the start.

7.1.3 HEATMAP BASED ON DURATIONS



All the activities have a colour in the heatmap that visually can give the impression of having all activities with a long duration. In truth, the fact is that in this phase all the tasks have a duration of 1s and, taking a look to the legend on the right, is possible to notice that red corresponds to this duration.

8 Comparison between Disco and Apromore (Bello Frati)

Mining the transition map both on Disco and Apromore we notice this:

- the activities are the same.
- Disco and Apromore considers different types of frequencies by default. Disco uses total frequency (absolute) that counts the total times a token passes on the arrow or on the activity while Apromore uses the Case frequency that counts the number of different tokens that passes on them.
- Is possible to change the type of frequency for transition map. Doing so, the results become the same.
- Changing the percentage related to path visibility, we observed a different behaviour, anyway for a better comparison we set this percentage to 100 so the maps shown on both platforms were the same using the same configuration.

Generally, when keeping the standard settings, Absolute Frequency on Disco and Case on Apromore, the behaviour is the following: in Disco when there some loop on the number of tokens that went back and go through an activity again is added to the number of tokens who have already been there, so higher numbers with respect to the number of tokens. In Apromore this doesn't happens, the number that can pass over a link is simply the number of tokens divided by the probability that a token goes through that link.

Another observation is that on both Apromore and Disco two flows disappeared with respect to the normative model simulation. These were:

- flow from "MLEngineer check test results" to "MLEngineer evaluate classifier".
- flow from "MLEngineer check test results" to the default gateway.

The reason for this is that, with the gateways set each at 50% probability, on some tasks with few tokens these simply didn't pass through those paths but continued with the other available. To solve this issue, we set the number of tokens to 140 and we modified some percentage probabilities (*total # of label sufficient?* 80% *YES*, 20% *NO* and *Evaluation phase?* 80% *YES*, 20% *NO*) to have at least one token that passes on each path.

Observing the transitions maps, the result we obtained is coherent with the parameter we set creating the simulation on BIMP, particularly:

- the number of total tokens is correct cause still the same we set in BIMP
- observing the most executed tasks they are coherent with the heatmap we produce on BIMP

To replicate the results the following is needed:

Disco

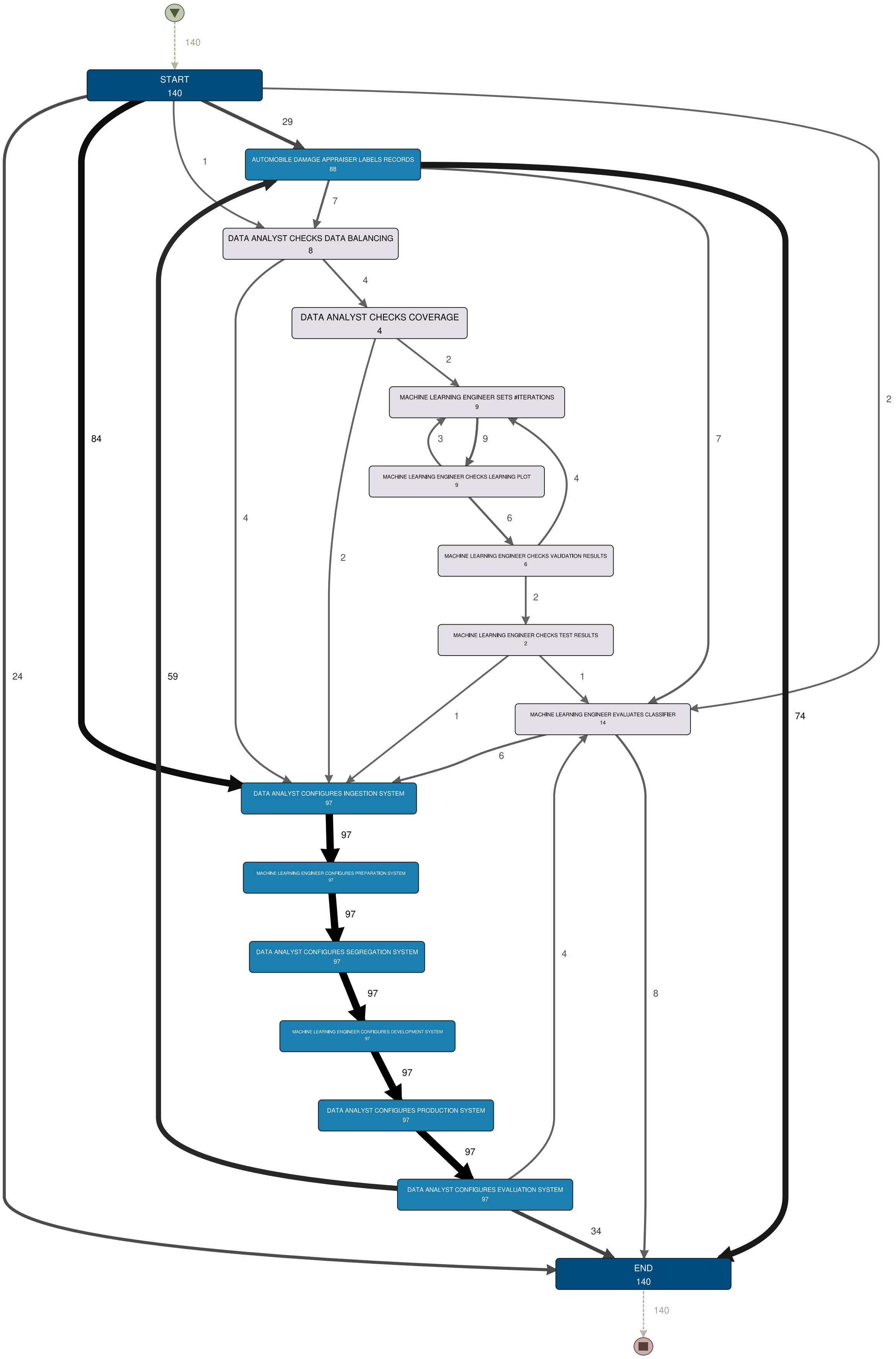
1. Import the "modified probabilities simulation_logs.mxml"
2. Set paths to 100%

Then the transition map can be exported in pdf with the "export" button (bottom right of the screen), and by selecting "process map".

Apromore

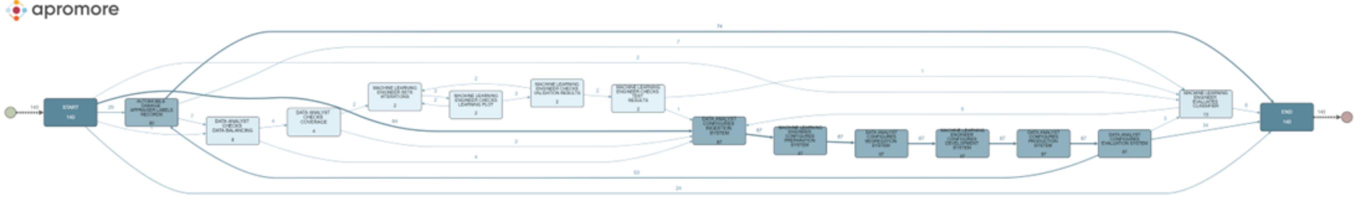
1. Import the "modified probabilities simulation_logsmxml"
2. Open the imported file by double clicking it
3. Set Arcs to 100%

Then the transition map can be exported with the "download as pdf" button on the top left of the screen.



140
START
140
29
1
84
24
AUTOMOBILE DAMAGE APPRAISER LABELS RECORDS
88
7
DATA ANALYST CHECKS DATA BALANCING
8
4
DATA ANALYST CHECKS COVERAGE
4
2
MACHINE LEARNING ENGINEER SETS #ITERATIONS
9
3
9
4
MACHINE LEARNING ENGINEER CHECKS LEARNING PLOT
9
6
MACHINE LEARNING ENGINEER CHECKS VALIDATION RESULTS
6
2
MACHINE LEARNING ENGINEER CHECKS TEST RESULTS
2
1
1
MACHINE LEARNING ENGINEER EVALUATES CLASSIFIER
14
6
2
7
74
59
4
2
DATA ANALYST CONFIGURES INGESTION SYSTEM
97
97
MACHINE LEARNING ENGINEER CONFIGURES PREPARATION SYSTEM
97
97
DATA ANALYST CONFIGURES SEGREGATION SYSTEM
97
97
MACHINE LEARNING ENGINEER CONFIGURES DEVELOPMENT SYSTEM
97
97
DATA ANALYST CONFIGURES PRODUCTION SYSTEM
97
97
4
8
DATA ANALYST CONFIGURES EVALUATION SYSTEM
97
34
END
140
140

apromore



9 Comparison between Apromore and ProM (Capecchi Niccolai)

9.1 preliminary steps and mining

To mine, some preliminary steps are required to obtain a CSV version of the log and to import it:

- ***Conversion of the log in CSV using Disco***
    1. From Disco, with the steps in the previous chapter we obtained its corresponding transition map
    2. From that page, clicking on "export" on the bottom right of the screen, a menu appears
    3. Select "event log"
    4. Select "export log as" → "CSV"
    5. Leave blank the available options and click on "export CSV files…"
    6. Save it locally

- ***Import and mining of the CSV in ProM***
    1. In ProM, select "import" from the home page, then select the CSV log exported from Disco
    2. Click on it from the main menu, then click the "play" button on the right
    3. Convert it to XES by selecting the "Convert CSV to XES" plugin.
    4. Select "case ID" as case column by clicking the "+" and for completion timestamp the "Complete Timestamp"
    5. Click on "next" then on "finish". Now a XES file is available in the main menu for mining
    6. Select it and click the play button
    7. From the "action" menu, select the "BPMN miner" plugin. Leave "inductive miner" as a miner algorithm
    8. De-select all attributes for primary key detection and click on finish
    9. Now, the BPMN can be found in main menu. It can be used in ProM directly or exported to disk

- ***Import and mining of the CSV in Apromore***
    1. From the APROMORE file system page, import the CSV log
    2. Double click it. The "process discoverer" page opens
    3. Toggle "BPMN model" on the top left of the screen
    4. Abstraction settings → arcs = 100%, to make sure that all the arrows are visible
    5. Select "save BPMN model" on the top left of the screen
    6. Now, the BPMN can be found in the APROMORE file system, and can be downloaded

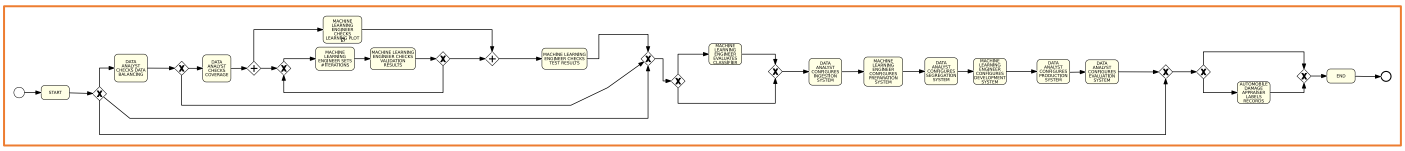
9.2 Differences between the models

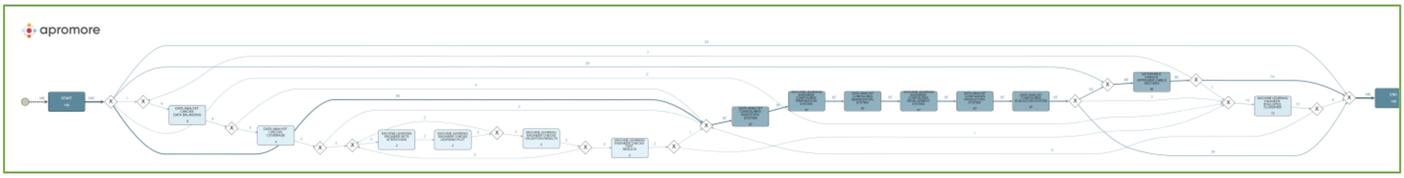
We notice that the mined BPMNs generated using the default settings were different.

At this point, with the parameters correctly set we observe that:

- The number of activities were the same in both diagrams
- On the BPMN mined on Apromore, there were more arrows and more gateways, indicating an higher complexity

Let's see the mined diagrams:



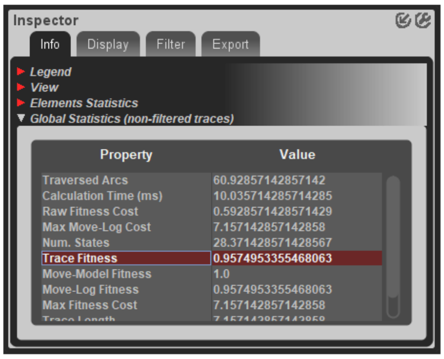


Comparing each mined diagram with the normative model we observed that:

- In the BPMN diagram mined on ProM a couple of parallel gateways, not present before, is introduced. The two parallelized branches are composed in this way: on the first branch there is the "… checks learning plot" task and on the second one there are "… sets #iterations" and "… checks validation results". Instead, in the normative, the order is mandatory to execute "… sets #iterations" and "… checks learning plot" (these two tasks can be repeated multiple times, until a fine number of iterations is found) before "… checks validation results" task.
- In the BPMN diagram mined on ProM, the "evaluate classifier" task is executed always before all the configures tasks. Both in Apromore mined diagram and in the normative model the evaluate classifier is for sure executed after the configuration tasks (if they are required).
- Both on mined diagrams the tasks that involves the automobile damage appraiser is placed near the end event. The difference is that in Apromore is possible to return back to previous tasks, while in ProM it's executed as the last task.
- In ProM after the configure phase the only executable task is the one related to the appraiser.

10 Conformance checks (Capecchi Frati)

10.1 BPMN mined on ProM



Property	Value
Traversed Arcs	60.92857142857142
Calculation Time (ms)	10.035714285714285
Raw Fitness Cost	0.5928571428571429
Max Move-Log Cost	7.157142857142858
Num. States	28.371428571428567
Trace Fitness	0.9574953355468063
Move-Model Fitness	1.0
Move-Log Fitness	0.9574953355468063
Max Fitness Cost	7.157142857142858

Precision : 0,83723

Generalization : 0,98395

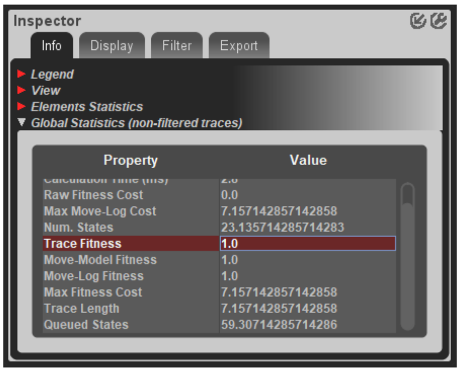
Trace fitness of the BPMN mined with ProM: 0.957

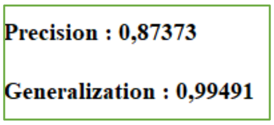
Precision is 0.83

Generalization is 0.98

Complexity of the mined BPMN on ProM: $\sum \#gateways + \#sequence\ flows + \#activities$: 12 + 34 + 16 = 62

10.2 BPMN mined on Apromore





Trace fitness of the BPMN mined with Apromore: 1.0

Precision is 0.87

Generalization is 0.99

Complexity of the mined BPMN on Apromore: $\sum \#gateways + \#sequence\ flows + \#activities$: 15 + 43 + 16 = 74

10.3 Comparison

The lower performance offered by ProM are justified by the fact that there're no paths that turns back to the configuration phase from successive tasks, while in Apromore these paths are present. The counterpart is that ProM is simpler, with a simplicity value that is lower (remember: *simplicity = #sequence flows + #activities + #gateways*).

Apromore has a trace fitness of 1.0, this means that the log is completely replicable on the mined BPMN.

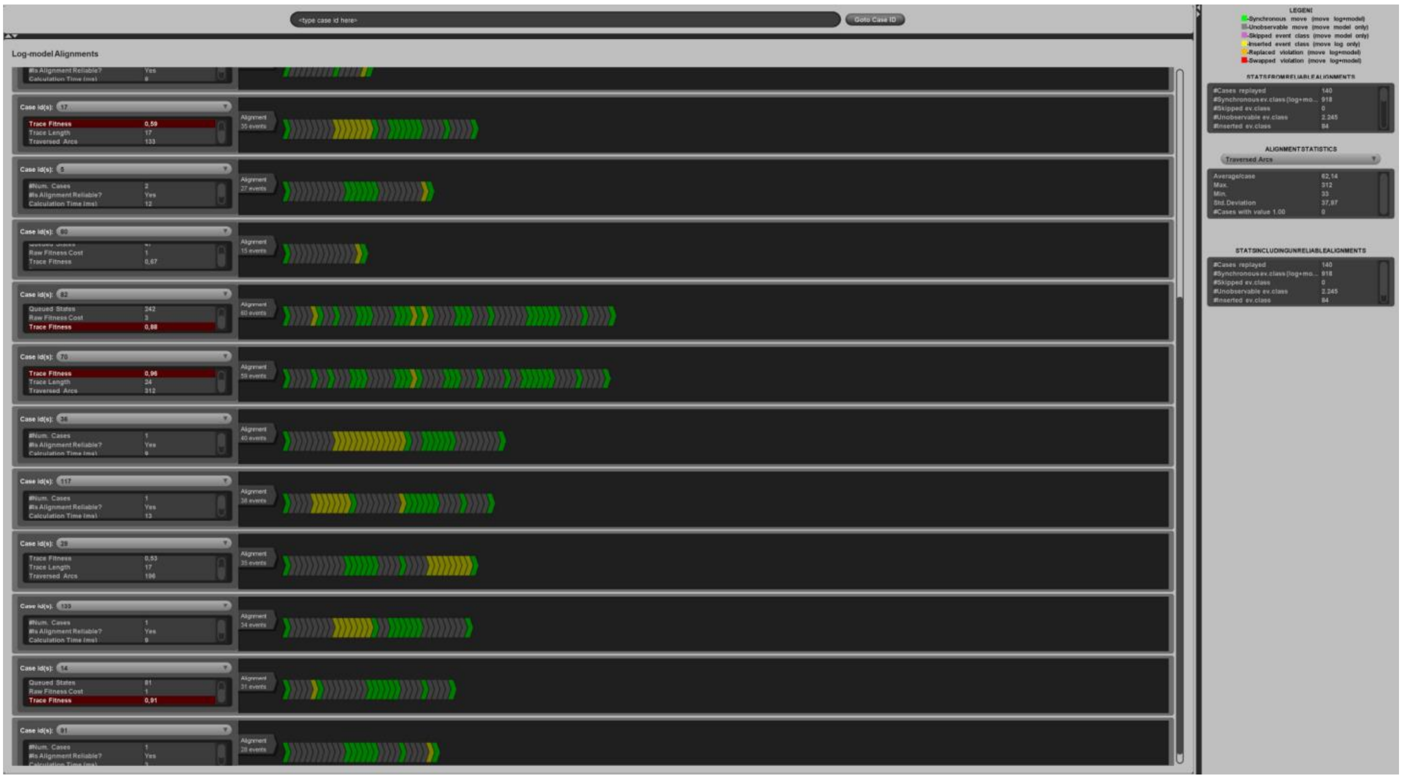
ProM has a trace fitness of 0.957 that is quite high too.

In both BPMN models the generalization is really high (0.99 and 0.98), it means that the models are not overfitting, and that the behavior isn't restricted to just the log. The precision is quite high too and it ensures us that our models are not underfitting. A precise model is a model that doesn't allow for behavior very different from what was seen in the event log.

11 Log model alignments

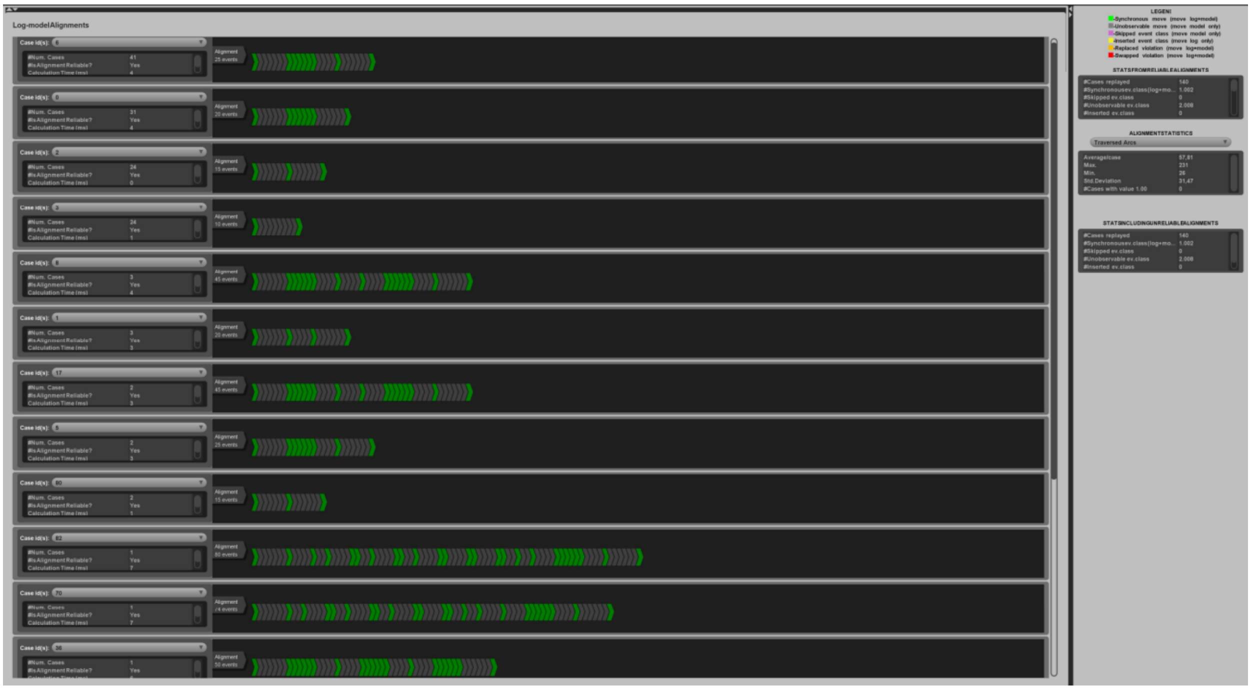
11.1 ProM (Capecchi, Frati)

Looking at the log model alignments, some groups of log moves (yellow colour) can be individuated. Their presence is due to the fact that in the ProM's mined diagram there're no returning back arrows, and this fact doesn't allow the model to run the configuration phase multiple times, while in the log there are some cases in which for different reasons a reconfiguration can be necessary. Probably this is the main reason why the trace fitness of the model has a value of 0.957 (lower than 1), but it's still a good result; the reason is that even if the log moves seems to be in a high proportion, in reality they're present only for the minority of the cases.



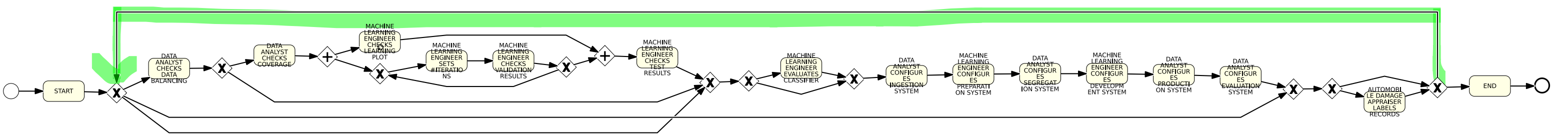
11.2 Apromore (Bello, Niccolai)

In the log model alignments below all the activity moves result as green coloured, that's because they correspond to moves reported on both the model and the log. So, for all the cases the trace fitness computed is equal to 1, it means that each of these log cases can be correctly replicated on the model. This is an expected result since the global trace fitness is equal to one.

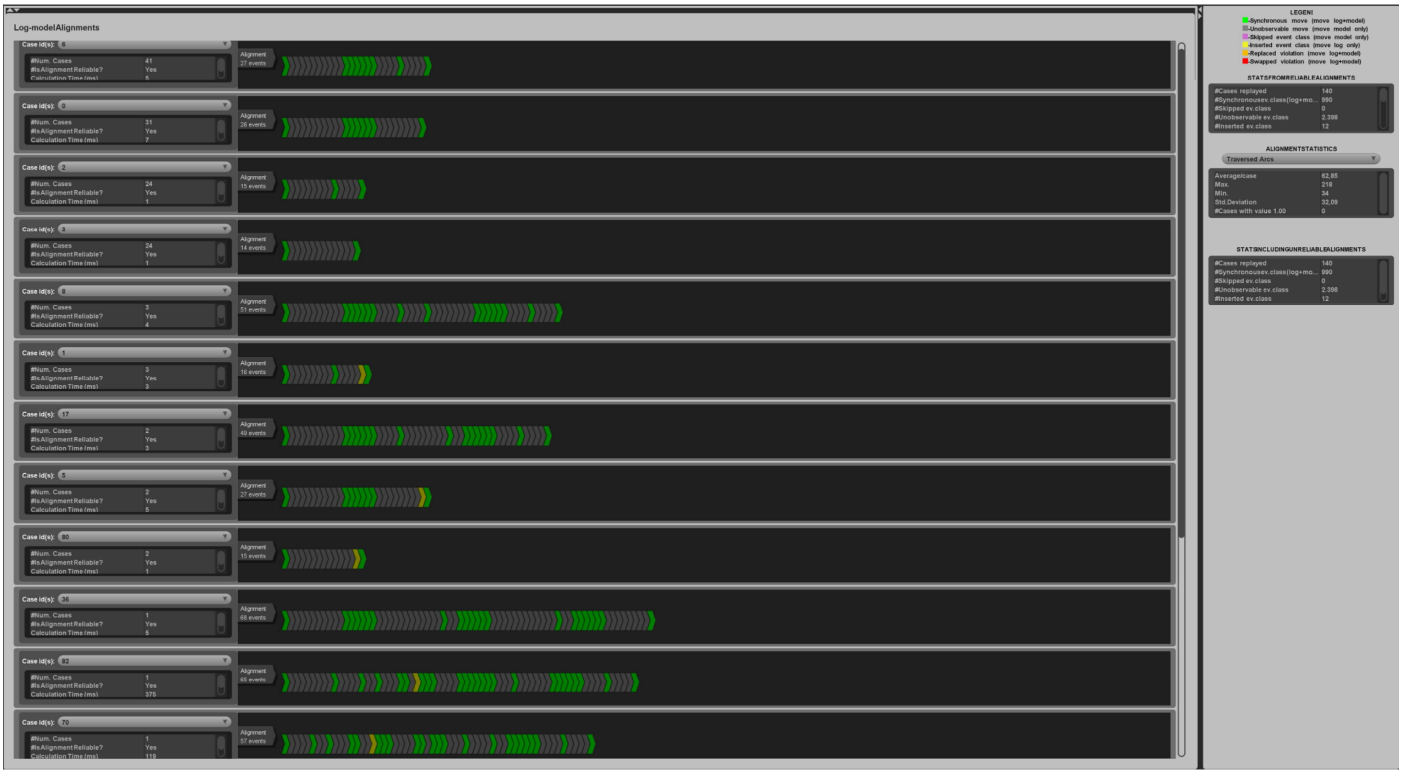


11.3 Test on the ProM mined model (Capecchi, Frati)

To confirm that the yellow log moves derived from the fact that in the ProM mined model there was an absence of gateways that consented to tokens to “go back” in the diagram, we decided to conduct a test on it; by comparing it with the normative BPMN, we strategically introduced an arrow (highlighted in green), that goes back to the first gateway, in order to consent this behaviour.



As we can see from the results obtained, the number of yellow log moves is reduced. This was just a test to confirm our theory and nothing more, since of course the next phases will be done by using the pure ProM mined model.



12 VIOLATIONS TO THE LOG (ALL)

In this phase three different violations have been inserted into the log. Each violation has been replicated in three cases, so at the end we have 9 violated cases. The three violations are:

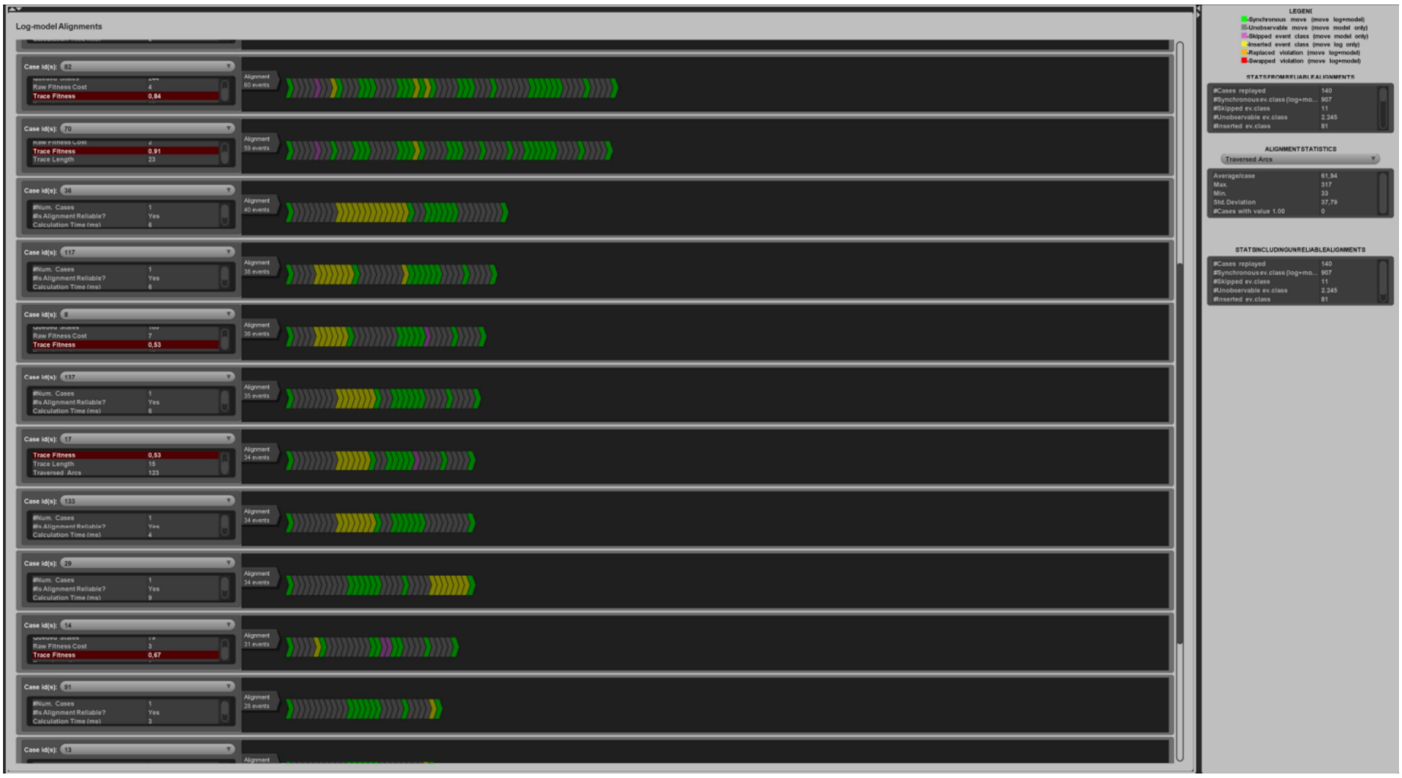
- Skipping of *Configure evaluation system* (cases 8, 17, 62). This is possible thanks to the assumption that this task is performed only in evaluation phase and not in all the others.
- Skipping of *Check data balancing* (cases 29,70,82). This is possible because in some determinate cases a factory could decide to acquire datasets from specialized data reseller. When an acquisition is made, the factory can ask for a desired distribution of data and we assume that resellers are trustable, so it’s not necessary to check the balancing of the classes in preliminary phase.

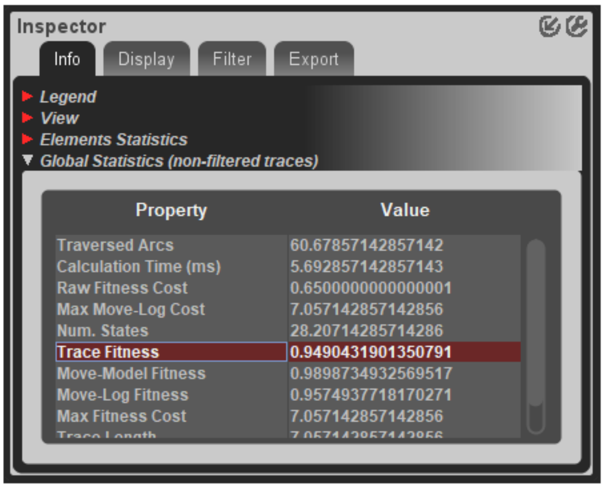
- Skipping of *Configure Segregation System* and *Configure development system* (5,9,14). This is possible because these systems are involved only in classifier development phase, so assuming that the classifier is already built up these are not mandatory.

To check if violations have an impact on the log, a simulation with the modified csv file has been made.

12.1 RESULTS ON PROM

As we can see, there are some model moves coloured in purple that correspond to skipped activities in the log. Checking which these are we see that there is a correspondence between the violations exposed above and these model moves. In case 70 and 80 the “check data balancing” task was skipped, then in case 17 the “configuration of the evaluation system” is visible as absent and finally in case 14 the subsequent skipped tasks are related to the third violation.

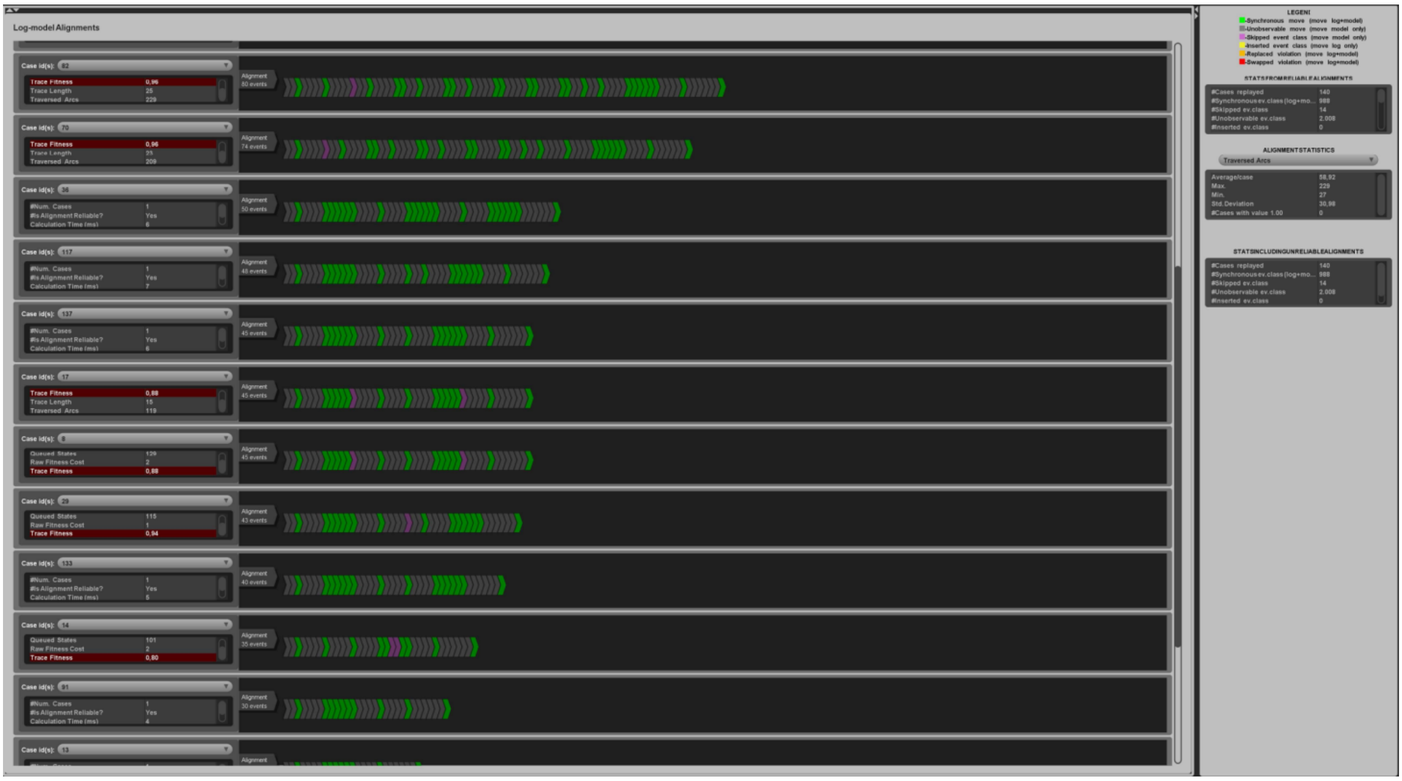


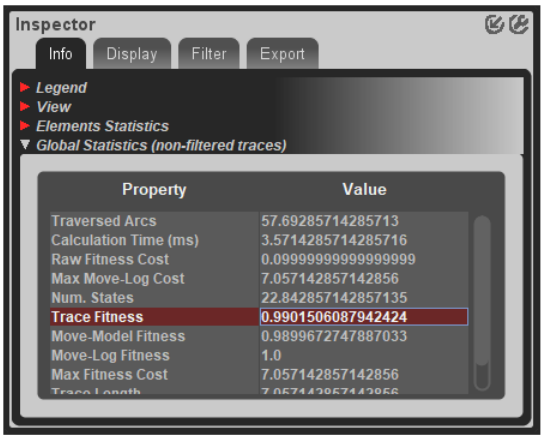


As we can see the trace fitness decreased but not in a substantial way (it decreased by circa 1%).

12.2 RESULTS ON APROMORE

In the Apromore log-model alignments, the violations are also visible. We can see the cases 29,82 and 70 with the violation of "*check data balancing*", then for 17 and 8 cases the skip of the "*configure evaluation system*", and in the end "*configure development system*" and "*configure segregation system*" are purple for case 14.

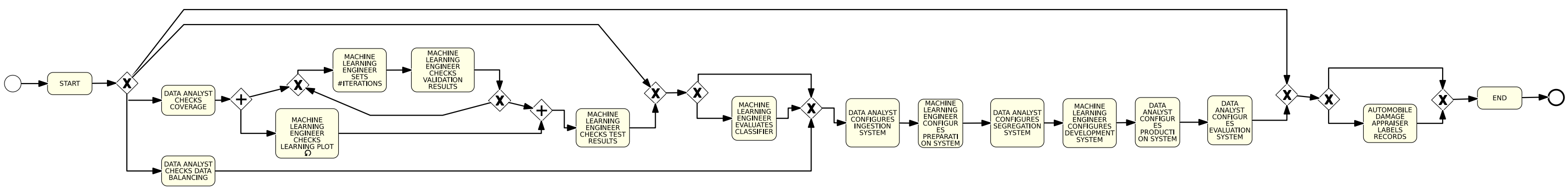




The fitness, as expected, goes below the precedent result; still, it remains very high. The reason for this is that we only modified 9 cases out of 140.

13 MINED MODELS ON THE MODIFIED LOG (Mining: Bello Niccolai, Conformance checking: Capecchi Frati)

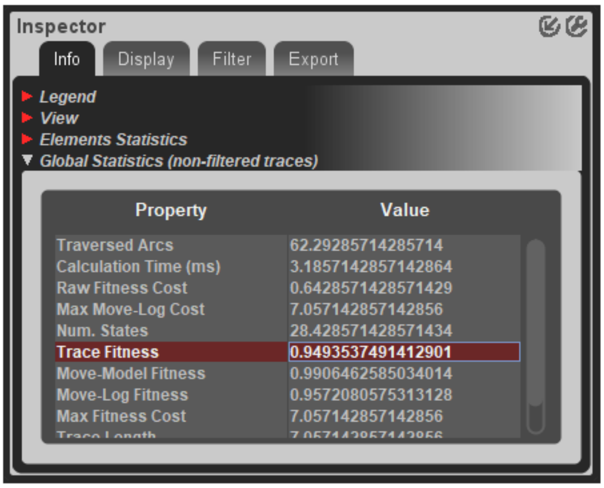
13.1 PROM



This is the BPMN mined by ProM on the violated log; as we can see, there aren't substantial differences with the normal mined model, except for the following:

- Now, the violation introduced to the "data analyst check data balancing task" brought the mining algorithm to separate it from the path relative to the development of the classifier. Essentially, if done, it goes straight on with the configuration of the system

Complexity of the mined BPMN on ProM: $\sum \#gateways + \#sequence\ flows + \#activities$: 11+31+16 = 58



Precision : 0,62183

Generalization : 0,99661

ProM results	Trace fitness	Precision	Generalization	Complexity
Mined BPMN on the actual log	0.95	0.837	0.98	12+34+16 = 62
Mined BPMN on the violated CSV	0.94	0.62	0.99	11+31+16 = 58

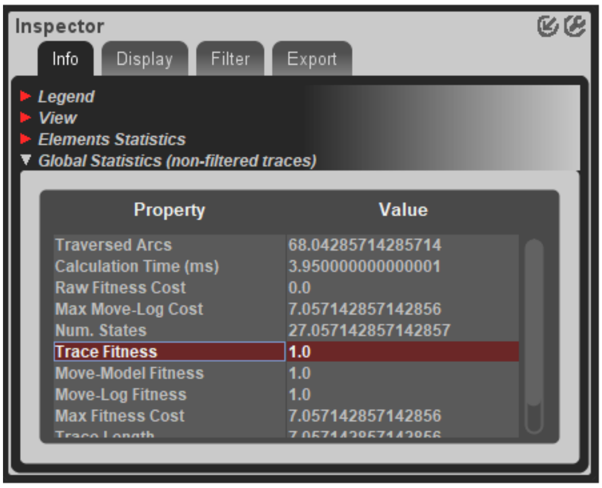
For the trace fitness, the number of gateways decreased by one and the sequence flows by three. Because of this, the new model is less complex, but not by much. The real price to pay comes in the terms of *precision*, since it got lower by the 20%; this quality dimension measures how much the behaviour of the discovered model can differ from what was seen in the event log. The obtained solution is not very precise because specific constraints on the behavior were removed by the mining algorithm. The rest of the results are comparable.

13.2 APROMORE

Comparing the BPMN mined on Apromore providing the modified log respect to the BPMN mined on Apromore without log modifications, we can observe that:

- About the tasks configuration sequence there is no difference in the order of execution, but in the new mined diagram some tasks can be skipped, and these are the configure for Segregation System combined with the configure for Development System, or only the configure for the Evaluation System.
- In the old diagram the execution of the "… data balancing" task was mandatory to the execution of the following task "… checks coverage", instead in the new mined diagram it is possible to skip the "… data balancing" task and execute only the "… checks coverage" one.

It is also interesting to highlight that the number of tokens that pass on the flow introduced to skip Segregation System and Development System configures corresponds exactly to the number of related cases edited in the modified log. The same thing happens when the Evaluation System is skipped.



Precision : 0,78887

Generalization : 0,99493

Apromore results	Trace fitness	Precision	Generalization	Complexity
Mined BPMN on the actual log	1.0	0.87373	0.99491	43 + 15 + 16 = 74
Mined BPMN on the violated CSV	1.0	0.78887	0.99493	41 + 15 + 16 = 72

The BPMN derived from the violated log has a lightly lower complexity with respect to the unviolated one. The trace fitness remains univariate equals to 1, while there is a very small change, almost negligible, in the generalization values. The biggest difference is in the precision value that decrease of around 9-10% because of violations.

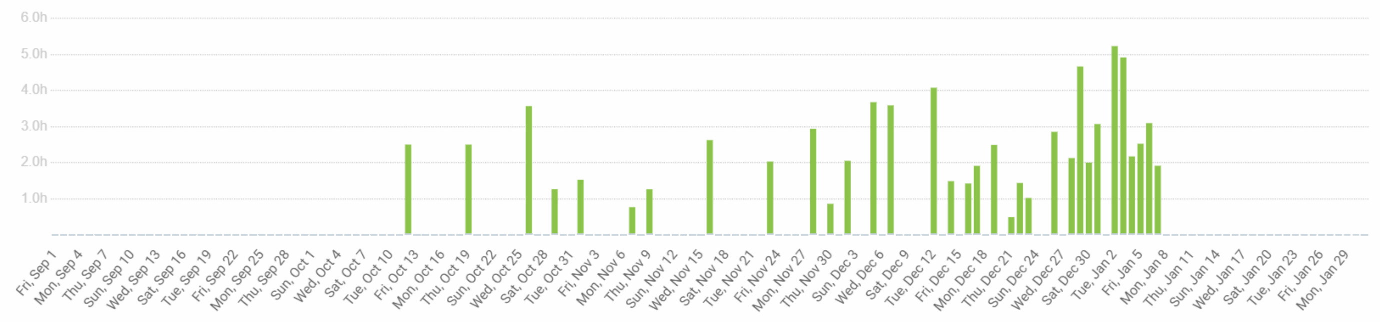
13.3 Overview

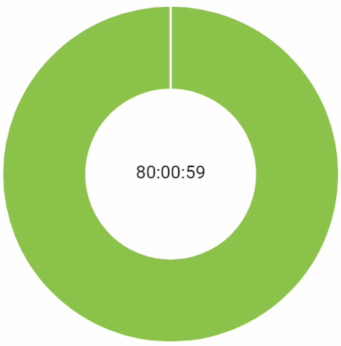
Model	Trace fitness	Precision	Generalization	Complexity
ProM + log	0.95	0.837	0.98	62
Apromore + log	1.0	0.87	0.99	74
ProM + violated log	0.94	0.62	0.99	58
Apromore + violated log	1.0	0.78	0.99	72

14 CLOCKIFY REPORT

14.1 BELLO

Total time	Top project	Top client
80:00:59	AUTOMATIC DAMAGES RECOGN...	

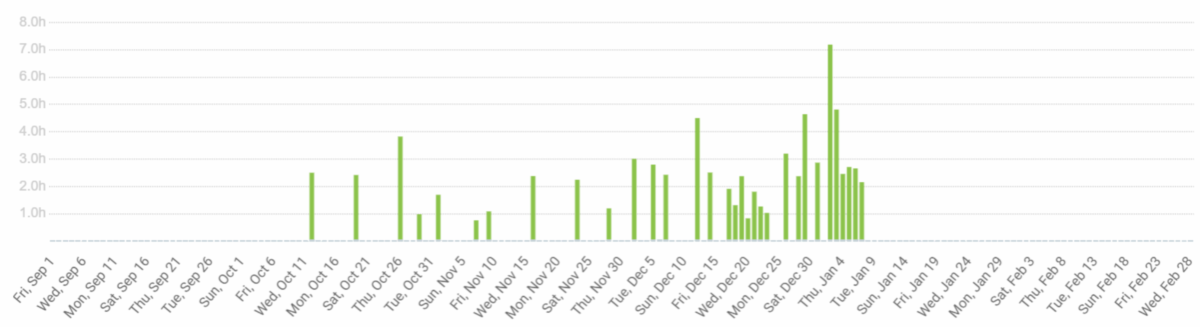


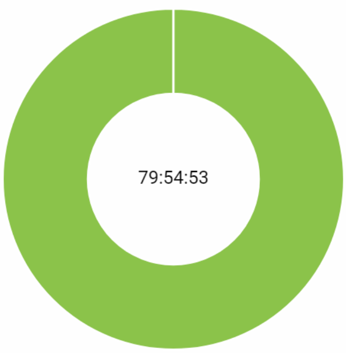


AUTOMATIC DAMAGES RECOGNITION 80:00:59 100,00%

14.2 CAPECCHI

Total time	Top project	Top client
79:54:53	AUTOMATIC DAMAGES REC...	

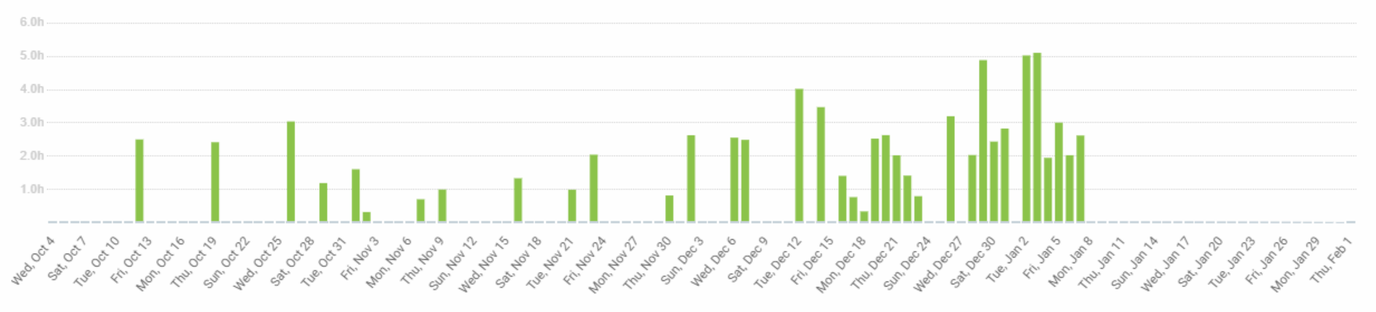




AUTOMATIC DAMAGES RECOGNITION 79:54:53 100,00%

14.3 FRATI

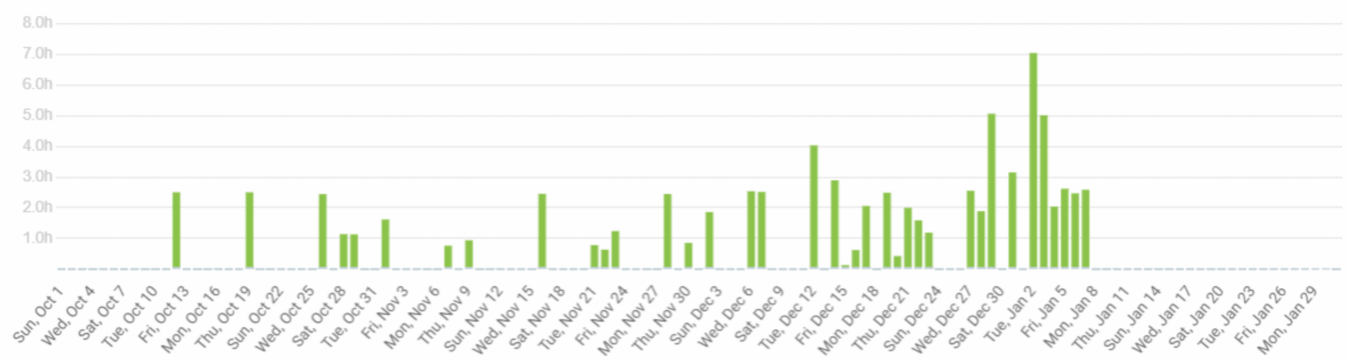
Total time	Top project	Top client
80:06:14	AUTOMATIC DAMAGES RECOGNITI...	

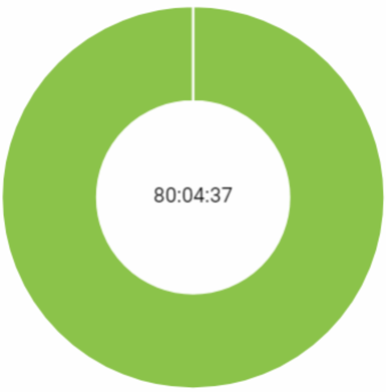


AUTOMATIC DAMAGES RECOGNITION 80:06:14 100,00%

14.4 NICCOLAI

Total time	Top project	Top client
80:04:37	AUTOMATIC DAMAGES REC...	





AUTOMATIC DAMAGES RECOGNITION 80:04:37 100,00%