Home task: KYC pass rate
Esmaili Emile – July 2021

# The Problem: The KYC pass rate slumped in Sept. '17

Each prospective Revolut customer has to go through a **Know Your Customer (KYC) process by submitting a photo ID and a picture to Veritas who perform 2 checks**:

• **Document check**: To verify that the photo ID is valid and authentic

• **Facial Similarity check**: To verify that the face in the picture is the same with that on the submitted ID.

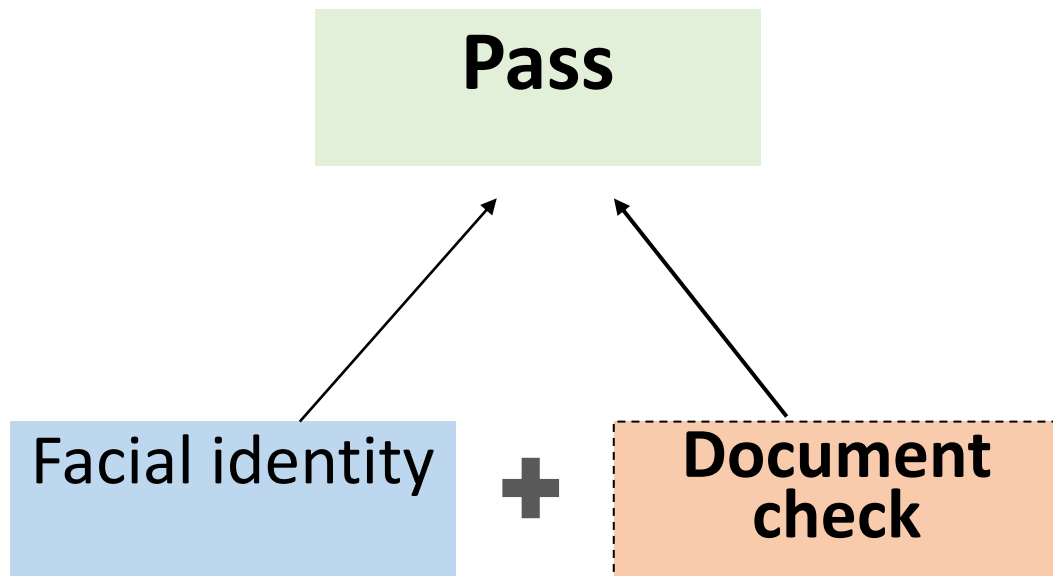**The customer will 'pass' the KYC process and get onboarded if the results of both checks are 'clear'**

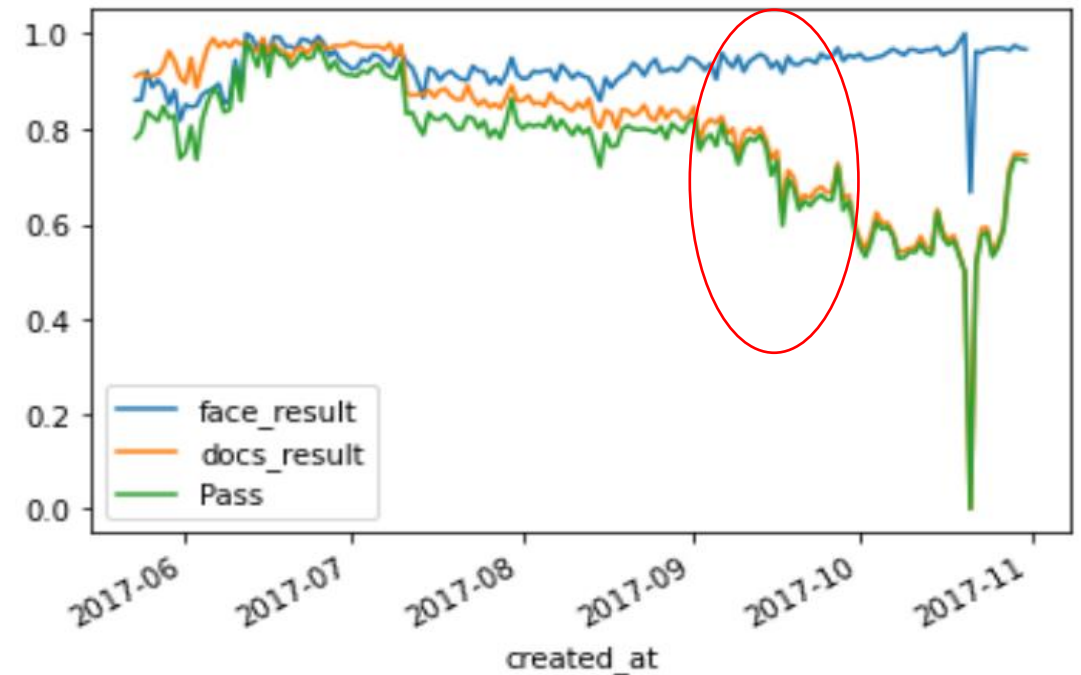⚠️ **The pass rate has decreased significantly**

~90%

~55%

June 2017

September 2017
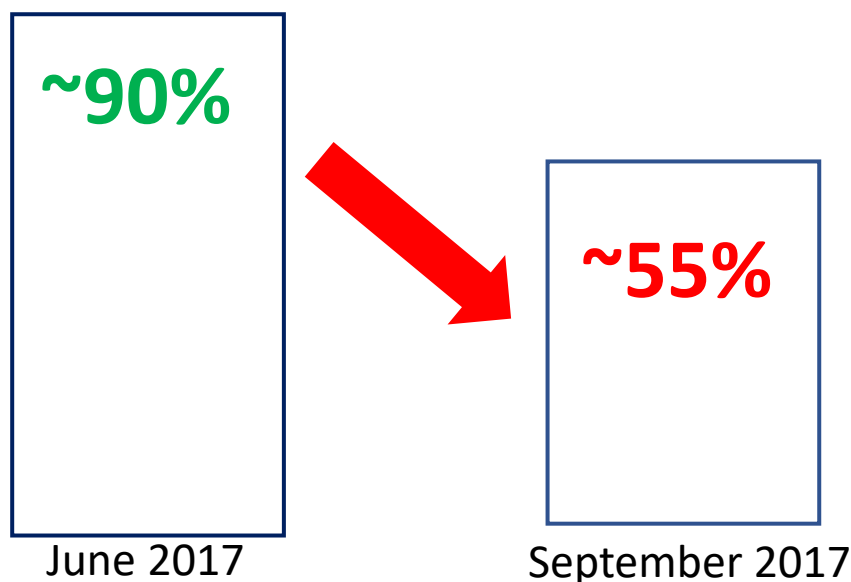
Reminder - Pass rate:

$$\frac{\# \text{ of customers who pass document \& face check}}{\text{number of customers who attempt}}$$

The main drivers of the decline - I

Pass

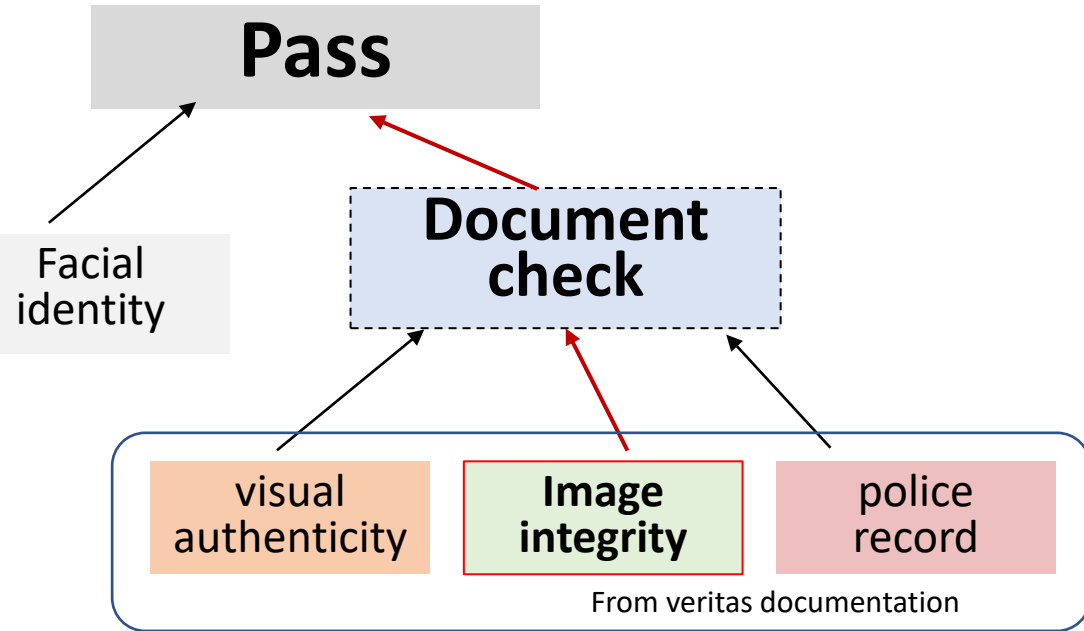Facial identity **+** **Document check**

Daily average pass rates, doc check and face check rates

The <u>document check</u> part of the process is the one at fault, we must investigate its components to determine the root

The main drivers of the decline - II

Pass

Facial identity

Document check

From veritas documentation

- visual authenticity
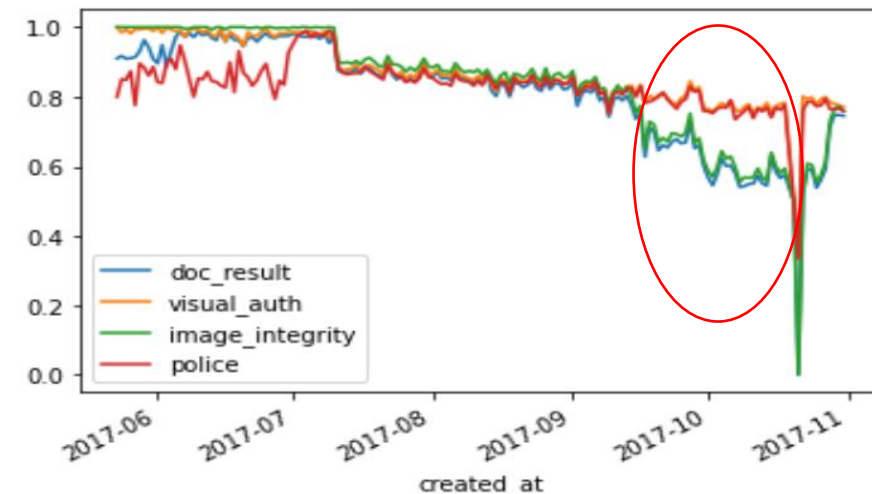- **Image integrity**
- police record

The image integrity check part of the process is the one at fault, we must investigate its components to determine the root
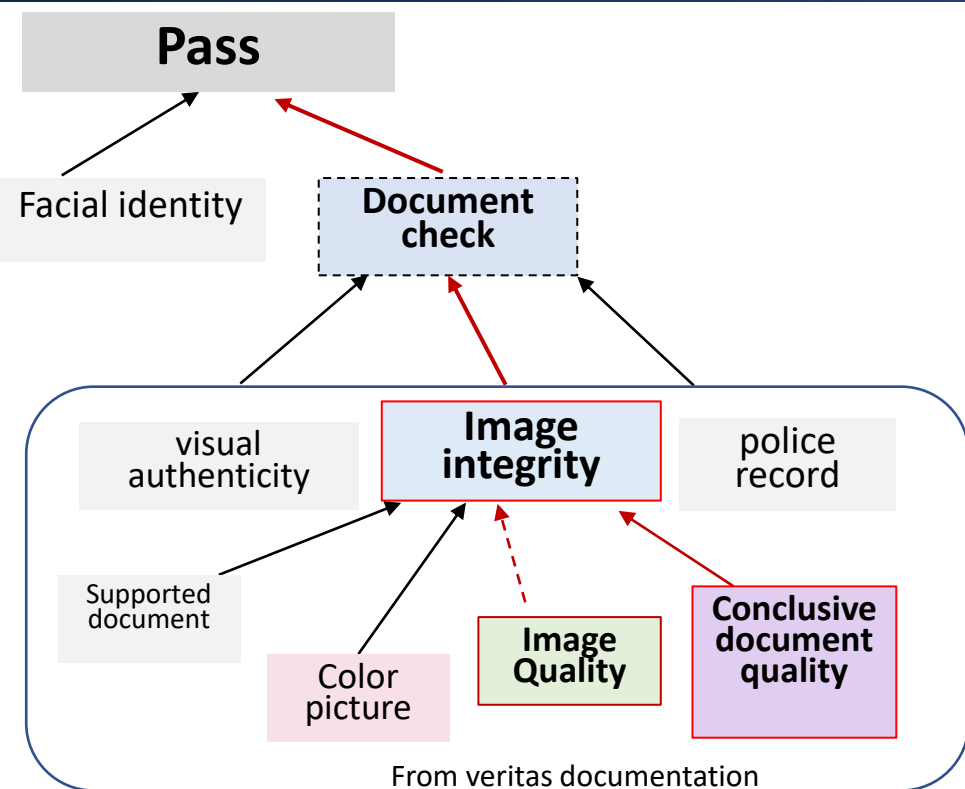
We use regression models to extract the sensitivity of the document check success rate to its components

Model: OLS regression of daily average document checks on their components		Model: Logit regression of intra-day document checks on their components (binary encoding)	
Variable	Sensitivity	Variable	Sensitivity
Visual authenticity	-0.099	Visual authenticity	5.88
Image integrity	**0.96**	**Image integrity**	**7.85**
Police record	0.11	Police record	-9.6

Visual inspection confirms both models' results:

The main drivers of the decline - III

Pass

Facial identity

Document check

visual authenticity

Image integrity

police record

Supported document

Color picture

Image Quality

Conclusive document quality

From veritas documentation

The conclusive document quality checks (and the image quality to a lesser degree) are to blame

We use regression models to extract the sensitivity of the image integrity success rate to its components

| Model: OLS regression of average daily image integrity checks on their components		Model: Logit regression of intra-day image integrity checks on their components (binary encoding)	
Variable	Sensitivity	Variable	Sensitivity
Supported document	-0.02	Supported document	-2.58
Image quality	**1.02**	**Image quality**	**5.97**
Color picture	0.11	Color picture	-8.37
Conclusive document quality	**0.96**	**Conclusive document quality**	**11.38**

Visual inspection

5
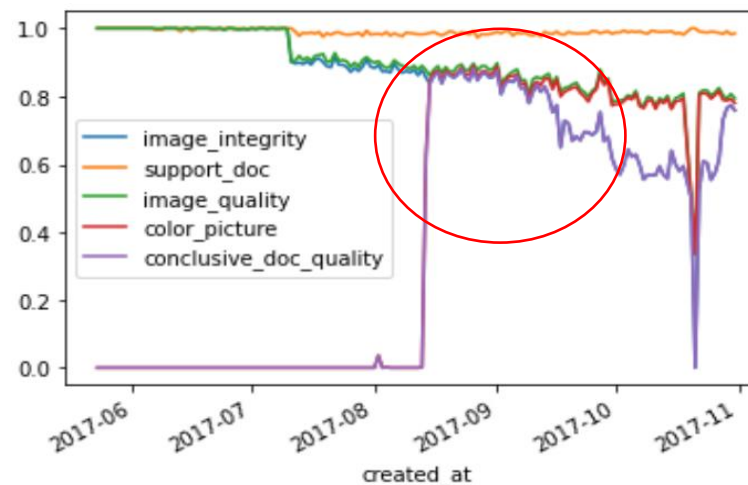
Working towards a solution: why so 'cautious'?

- Looking at the **subresults** for the two culprits, we can see that the main root cause, the Conclusive Document Quality result (CDQR), is **almost always assigned to a 'caution' error**, which according to Veritas means:

*"If any other underlying verifications fail but they **don't necessarily point to a fraudulent document** (such as the name provided by the applicant doesn't match the one on the document)"*

- It would be interesting to dig further with the help of Veritas Engineers as to why caution is assigned as opposed to "rejected" or "suspected"

Distribution of the subresults, when the check hasn't cleared
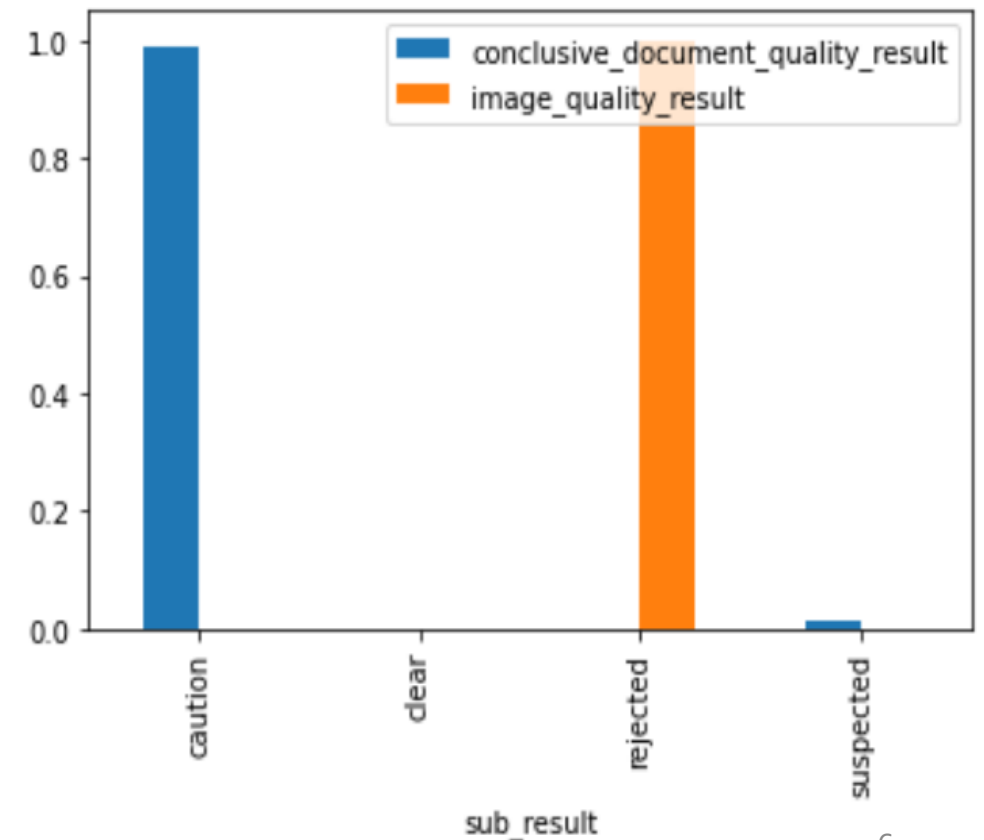
Conclusions

- The **Conclusive Document Quality Result** is the main driver behind the decrease of the pass rate: **the introduction of this metric coincides (with some lag) with the fall observed in early September 2017**

- As per veritas documentation "A result of clear in the conclusive_document_quality breakdown of image_integrity will assert if the document was **of enough quality to be able to perform a fraud inspection**"

Solutions

- Implement a **minimum resolution requirement** for the images uploaded

- **Work with Veritas engineers** to investigate the phenomenon of CDQR

- **Higher attempts** authorized per user (3 instead of 2)

- Find out **why subresults often assign 'caution'** for these errors