Building robust and defensible ML systems in Fincrime

Luca Traverso
Lead Data Scientist @Wise
Why Machine Learning.
Fincrime Team Objectives

- Prevent onboarding of criminals
- Minimizing the impact on good customers
In fast growing companies

*Backlog*: alerts / reviews generated
In fast growing companies

*Backlog*: alerts / reviews generated
In fast growing companies

Financial Risk

Customer Negative Impact

Time to alert
Machine learning system

Gives us:
❖ **Stable** backlog creation
❖ **Reduce** risk exposure

Needs to be:
❖ **Robust**: it adapts to changing conditions
❖ **Defensible**: understandable and credible to a third party
How do we do that?
ML system components

1. Feature creation & labelling
2. Automatic model retraining (closed-loop learning)
3. Model governance
4. Model interpretability
ML system components

1. Feature creation & labelling
   2. Automatic model retraining (*closed-loop learning*)
3. Model governance
4. Model interpretability
Features creation

Feature: The numerical representation of an observed behaviour

Defensible:
If features can be explained model results can be better interpreted / explained
Labelling

Tooling

State 0

State 1

State 2

State 3
ML system components

1. Feature creation & labelling
2. **Automatic model retraining** *(closed-loop learning)*
3. Model governance
4. Model interpretability
Automatic model retraining

- Features monitoring
- Features gathering
- Labels
- Training sets
- Model training
- Model tuning
- Model training
- Model testing
- Cost matrix
- Threshold selection
- Model deployment
- Model scoring
- OPS review / decision

@ least daily
Frequently or based on drift
Automatic model retraining

- Features monitoring
- Features gathering
- Model Tuning
- Model Training
- Model deployment
- Model scoring
- Cost Matrix
- Model testing
- Threshold selection
- OPS review / decision

- Labels
- Training sets
- @ least daily
- Frequently or based on drift
Automatic model retraining

Features monitoring
Features gathering

Features
Labels
Training sets

Model Training
Model Tuning

Model deployment
Model scoring

Cost Matrix
Threshold selection
Model testing

OPS review / decision
Closed-loop learning

- **Features monitoring**
- **Features gathering**
- **OPS review / decision**
- **Labels**
- **Training sets**
- **Model Training**
- **Model Tuning**
- **Model scoring**
- **Model deployment**
- **Cost Matrix**
- **Threshold selection**
- **Model testing**

- **Features monitoring**
- **Features gathering**
- **OPS review / decision**
- **Labels**
- **Training sets**
- **Model Training**
- **Model Tuning**
- **Model scoring**
- **Model deployment**
- **Cost Matrix**
- **Threshold selection**
- **Model testing**
ML system components

1. Feature creation & labelling
2. Automatic model retraining (*closed-loop learning*)
3. Model governance
4. Model interpretability
Model governance

❖ Product coverage
❖ Risk typology coverage
❖ Data always up-to-date
❖ Update models frequently (no do-and-forget attitude)
❖ Tracking all important model metrics (in prod)
❖ Features & labels quality monitoring

ML system components

1. Feature creation & labelling
2. Automatic model retraining (closed-loop learning)
3. Model governance
4. Model interpretability
Model interpretability

Shapley or Lime for ‘local’ interpretability

```
0.79
```
Model interpretability

Shapley or Lime for ‘local’ interpretability

Features linked to a specific pattern or behaviour - easier to interpret/explain the model decision
Current challenges.
Challenges

1. Scaling of ML infrastructure (with growth)
2. Features group interpretability (generating narratives)
3. Enhancing features and labels quality monitoring
Thank you!