

Resilient Machine Learning

<https://dsc0.usfdatainstitute.org>, March 2023, San Francisco

Oliver Zeigermann

Slides: <https://bit.ly/dsc0-resiliency-2023>

PDF: <https://github.com/DJCordhose/ml-resources/raw/main/pdf/Resilient%20Machine%20Learning.pdf>

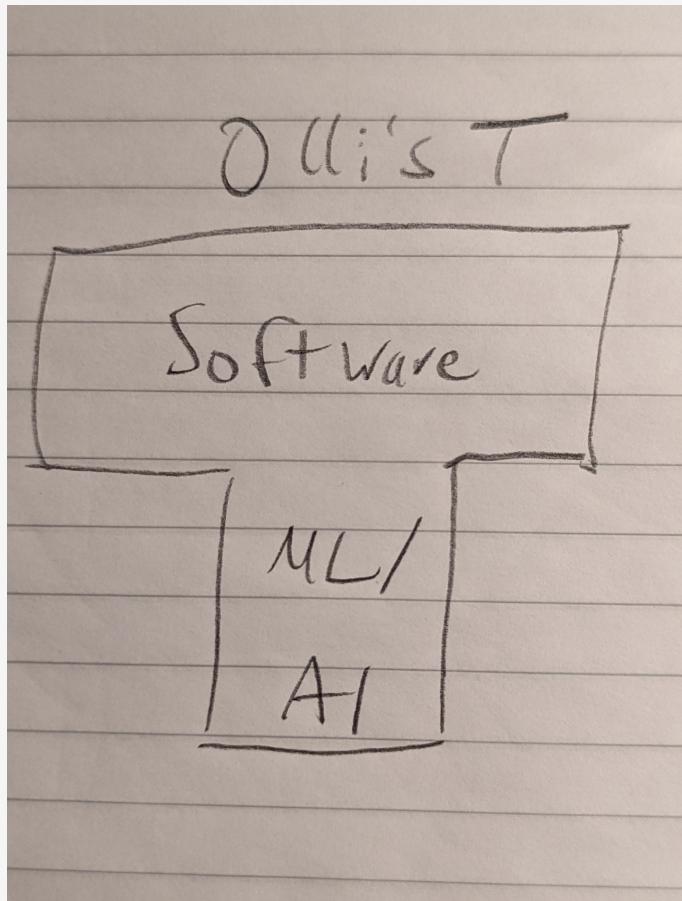
Gauge by show of hands

What do you think it most important in a machine learning project?

1. Have a high accuracy (or other relevant metric)
2. Use a novel/fancy approach
3. be able to explain what your model does
4. bringing something meaningful into production
5. know if it makes sense to bring something into production in the first place
6. have a way of knowing if a production model is still good (and knowing how to act upon that insight)

Please be as open as possible

Who is Olli



Oliver Zeigermann: Blue Collar Architect(ML)@OPEN KNOWLEDGE

Resilience

ability to adapt to difficult or unexpected situations

<https://en.wikipedia.org/wiki/Resilience>

Resilience in the world of Machine Learning

Dealing with Uncertainty

Agenda

1. managing uncertainty
2. deploying machine learning services
3. adversarial attacks and stability
4. drift and monitoring

Agenda

1. *managing uncertainty*
2. deploying machine learning services
3. adversarial attacks and stability
4. drift and monitoring

ML comes with a lot of uncertainty

- model and training
 - score
 - confidence
 - training vs test vs out of sample, real world
- will the approach work at all
 - depends on what "work" means
 - what score is good enough?
 - what about the other requirements

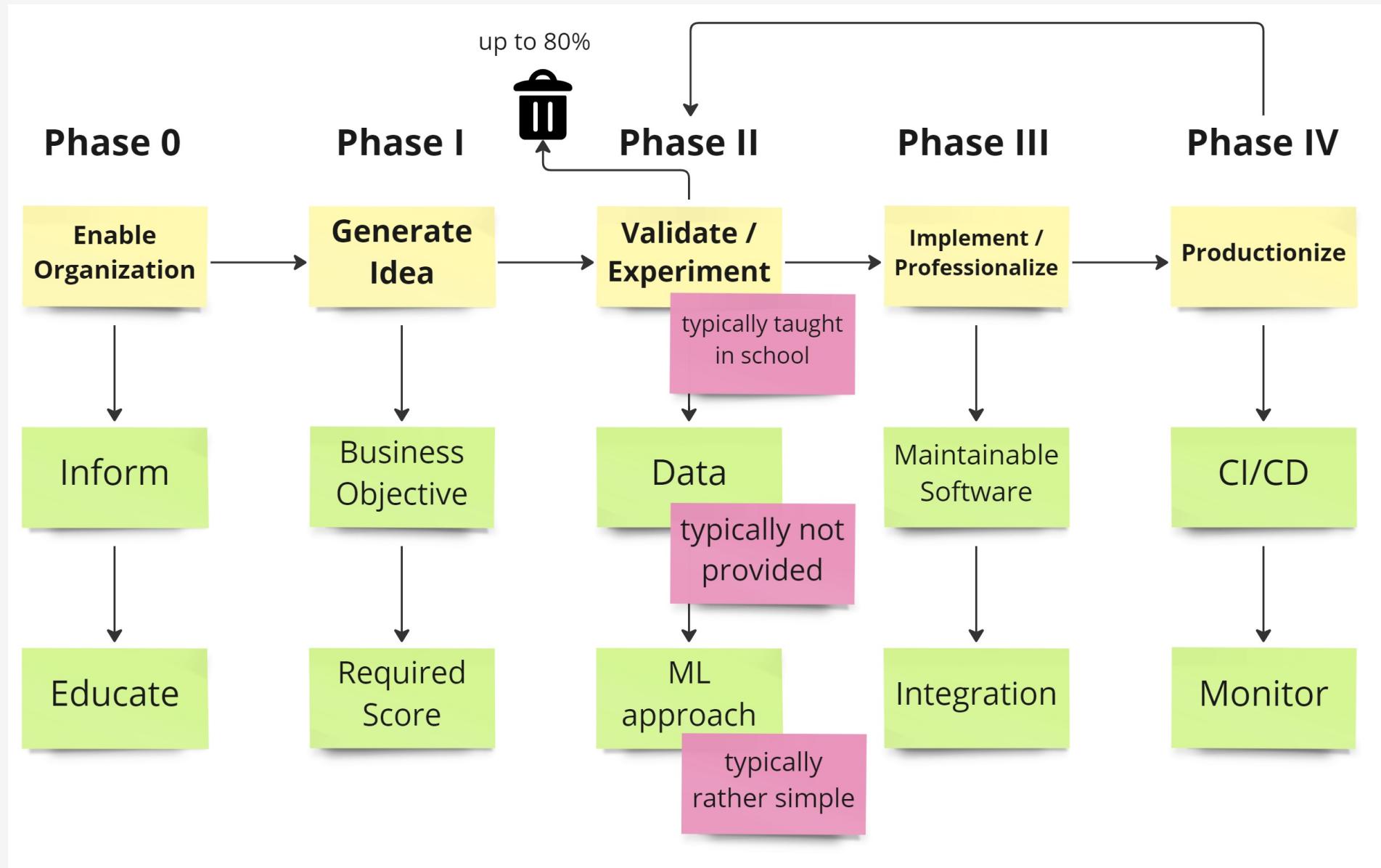
Uncertainty is hard to bear

- emotionally
- risk for business
- You need to manage the uncertainty / introduce resilient concepts to handle the stress
- also relevant for development process
 - create PoC, work in phases
 - try and infer life time of model

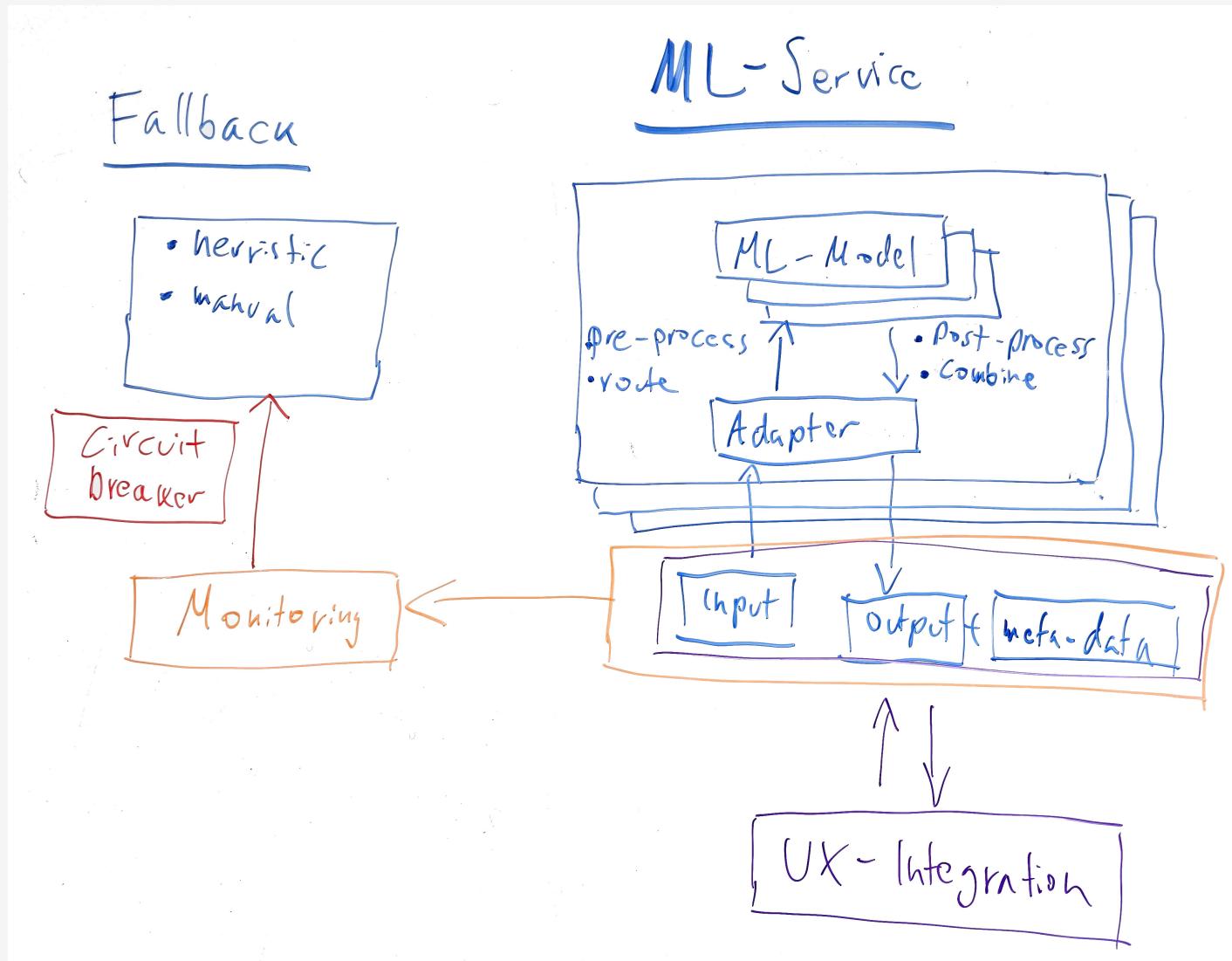
Agenda

1. managing uncertainty
2. *deploying machine learning services*
3. adversarial attacks and stability
4. drift and monitoring

Machine Learning Projects can be structured in phases



You don't just deploy the model



Agenda

1. managing uncertainty
2. deploying machine learning services
3. *adversarial attacks and stability*
4. drift and monitoring

Hacking the system / Adversarial Attacks



https://www.instagram.com/reel/CkQUhLov9_u/?igshid=MDJmNzVkJY=

Hacking the system is more common than you might think

taxes, laws, regulations, contracts, embargoes

- this is actually the job of a lot of people
- transparency vs hackability

Model stability and adversarial vulnerability

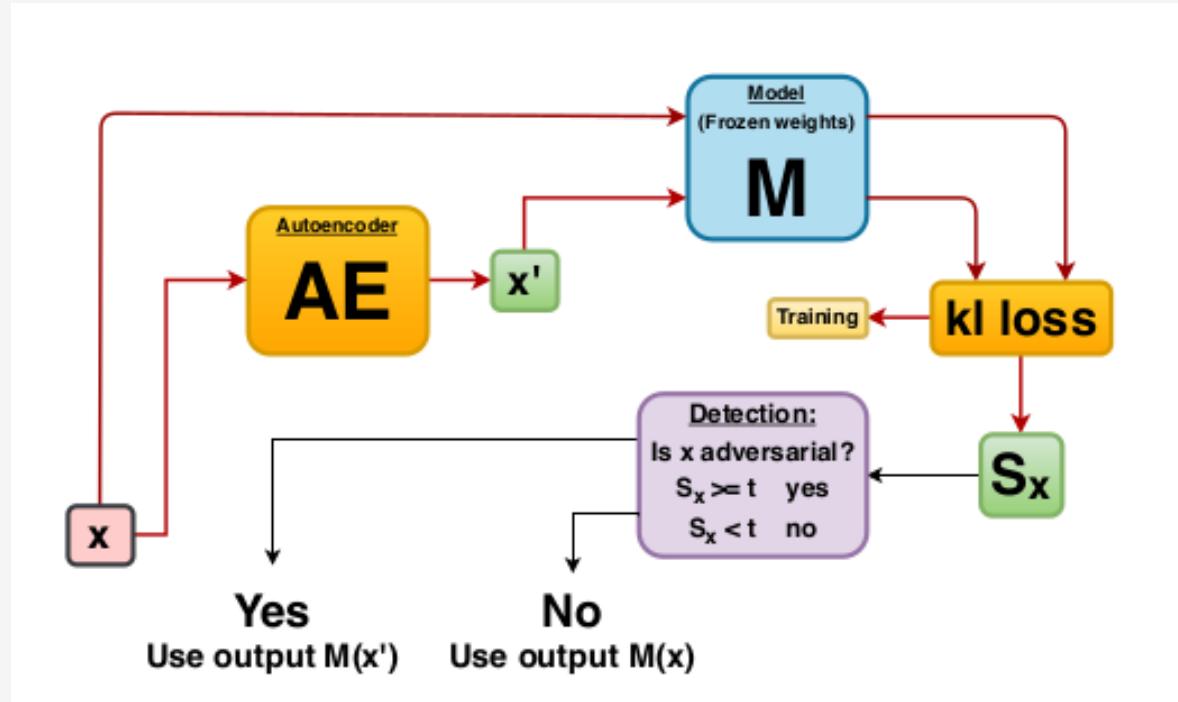
Main question: does slightly perturbing the input data yield a drastically different prediction?

If so

- there is an additional attack vector
 - because people could learn decision boundaries
 - by slightly tweaking features that do not require exact entry get an advantageous prediction
- high local variation hints towards undetected overfitting
- stability
 - high local variation makes it likely that retraining with new data will yield a completely new model
 - also requires new interpretation etc.
 - unwanted disruption for users

Adversarial AE detector

adversarial perturbation gives significantly different outputs at $M(x)$ and $M(x')$



still able to detect the adversarial examples in the case of a white-box attack where the attacker has full knowledge of both the model and the defence

<https://docs.seldon.io/projects/alibi-detect/en/stable/ad/methods/adversarialae.html>

<https://github.com/SeldonIO/alibi-detect#adversarial-detection>

Agenda

1. managing uncertainty
2. deploying machine learning services
3. adversarial attacks and stability
4. *drift and monitoring*

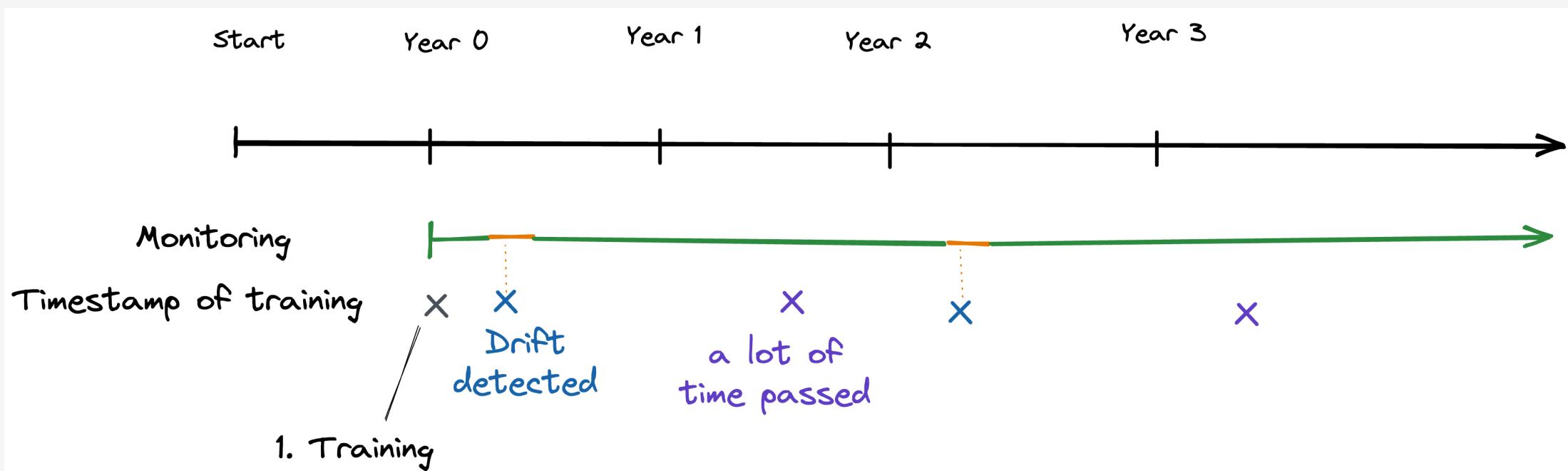
Machine Learning Systems are dynamic in nature



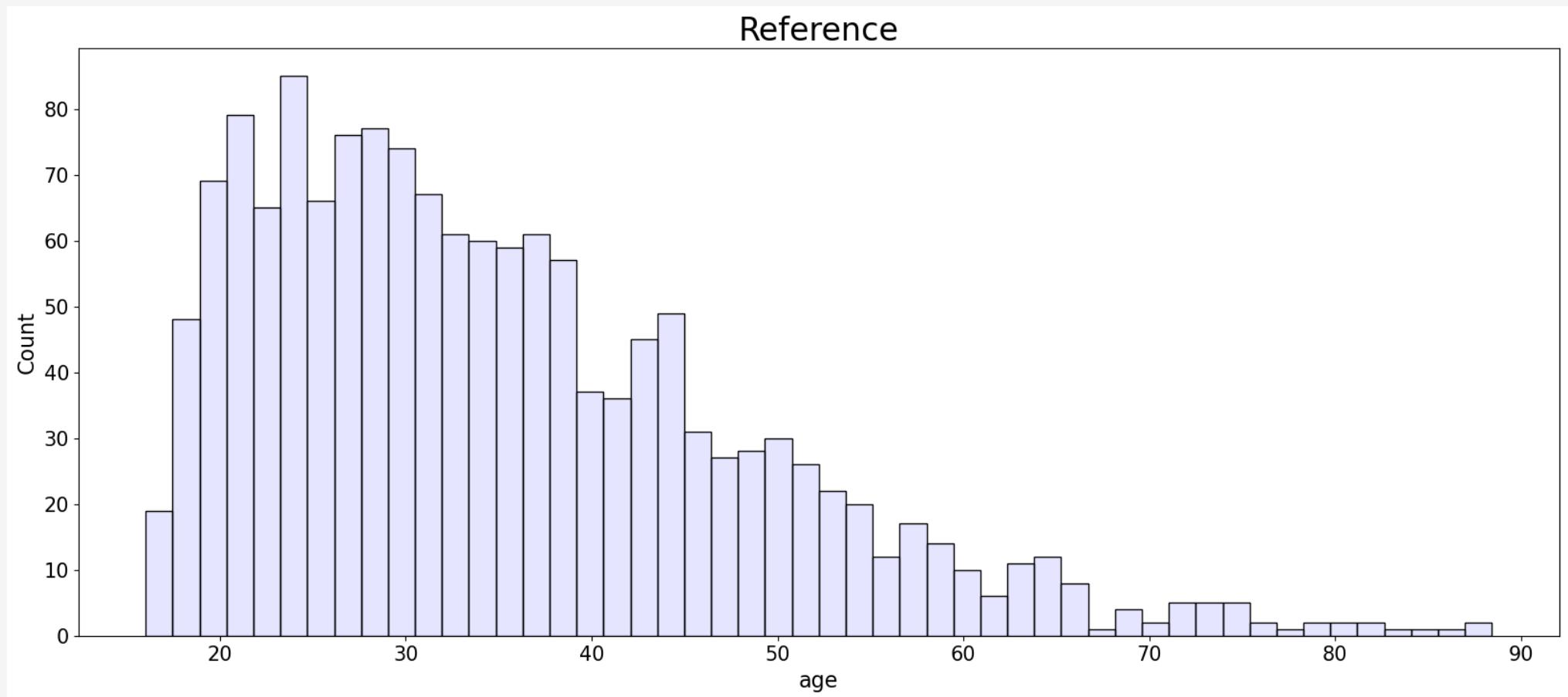
How do you know you need a new model in production?

- Check how the model behaves on newer data
 - How quickly does performance degrade?
 - check already in the exploration phase
- At least once a year, to make sure someone still knows how to even do this
- When the metrics degrade in production
 - ground truth from production data required to even find out
 - Sometimes you get this immediately after the prediction by the reaction of a human user
 - But often only after a significant delay
- *When the distribution of prediction data is significantly different from that of training*

At time of training?

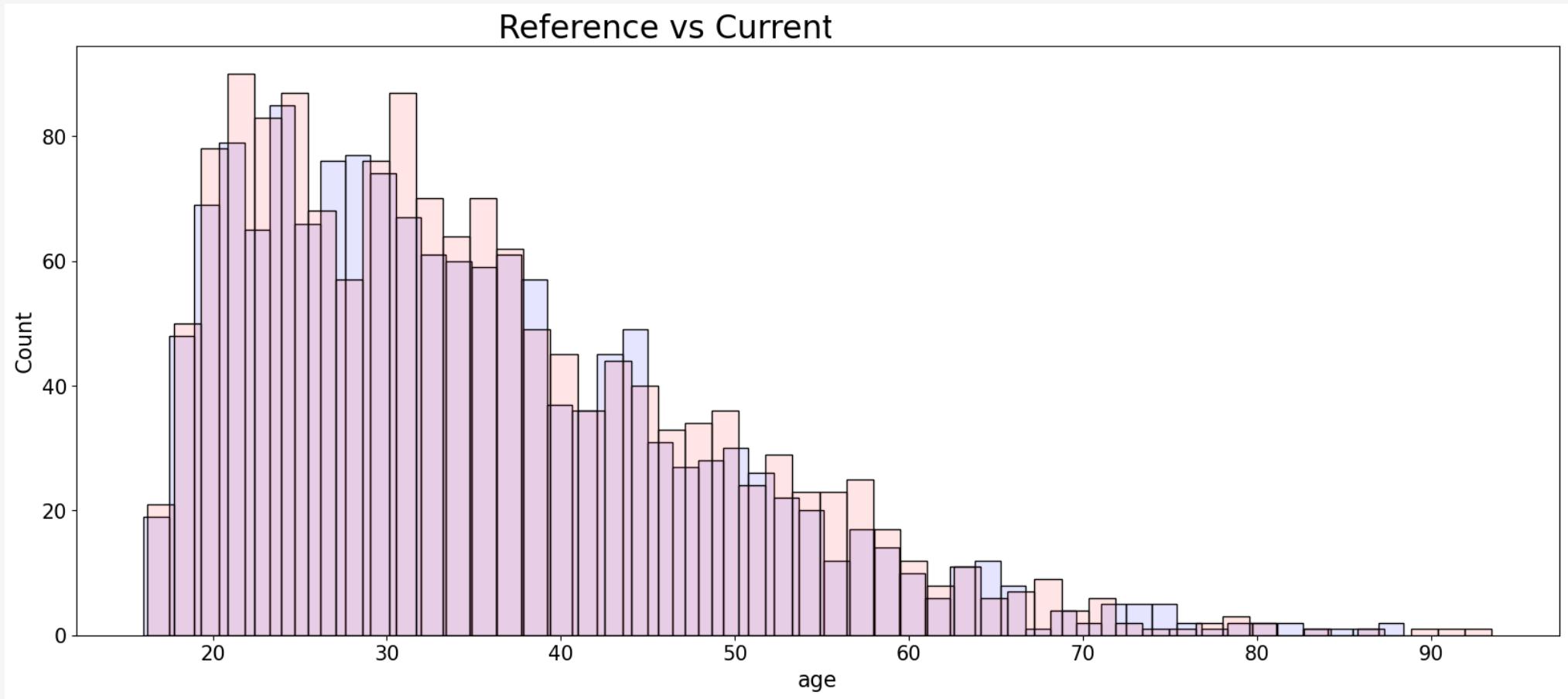


Reference Distribution

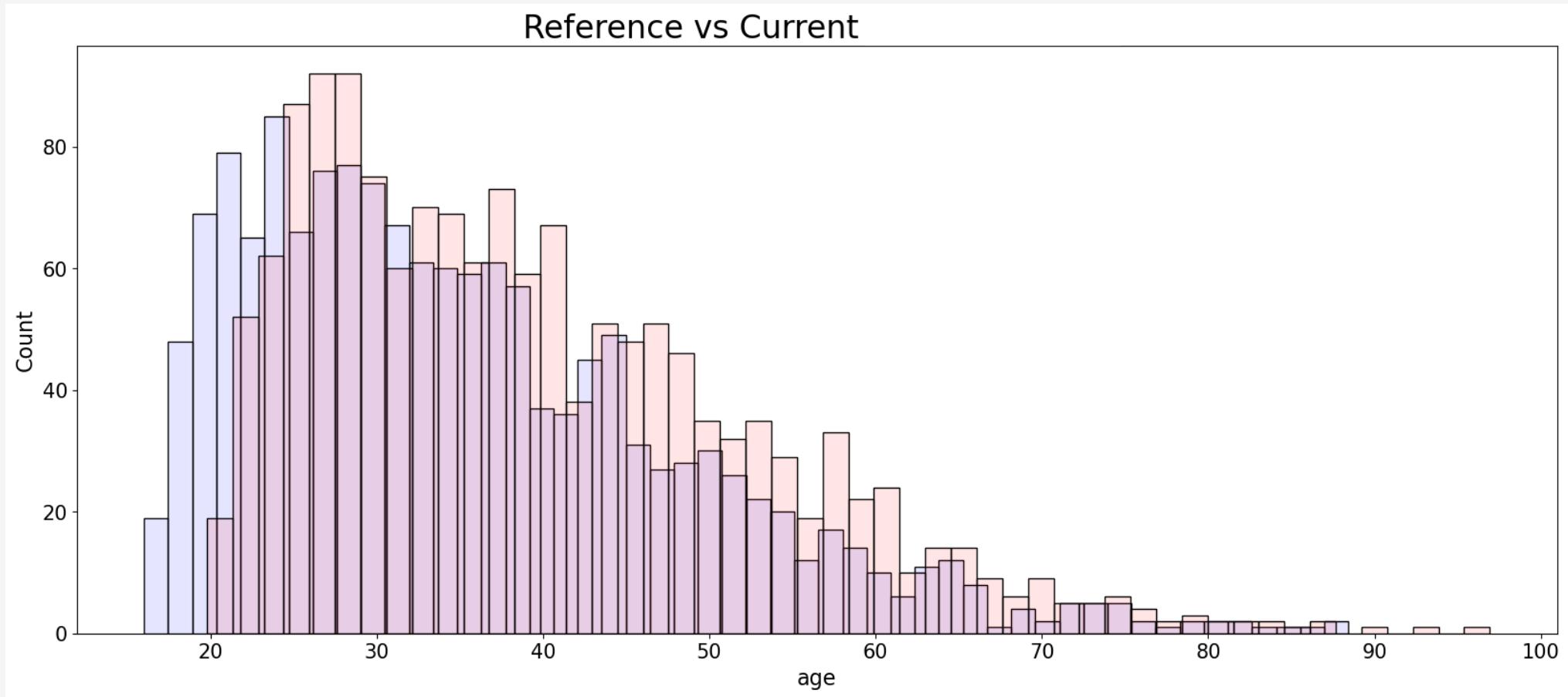


Example: Age of people seeking for insurance

Does it drift?



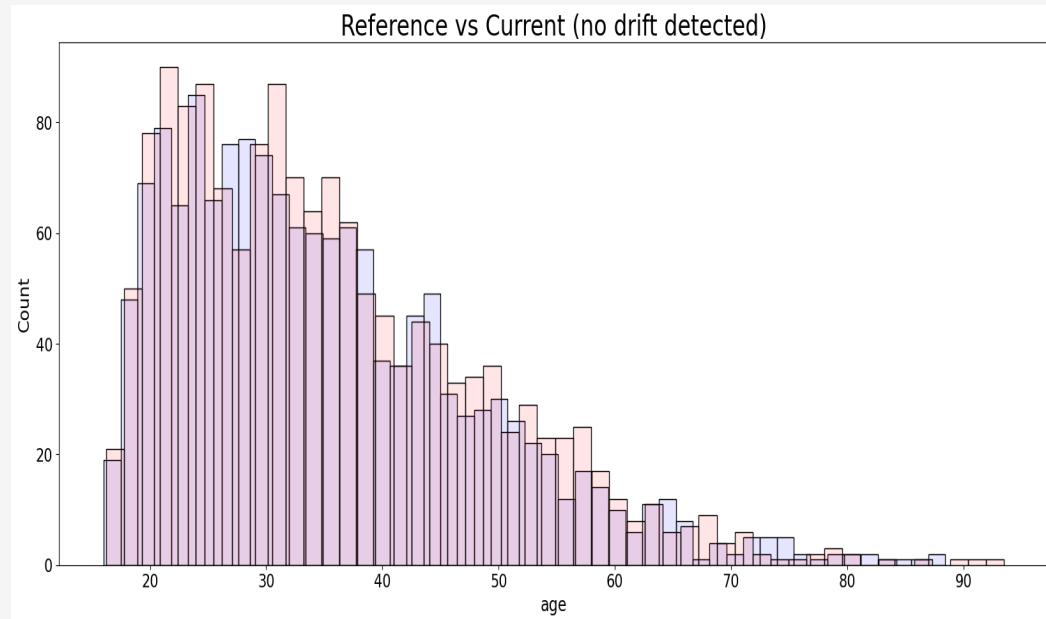
And this?



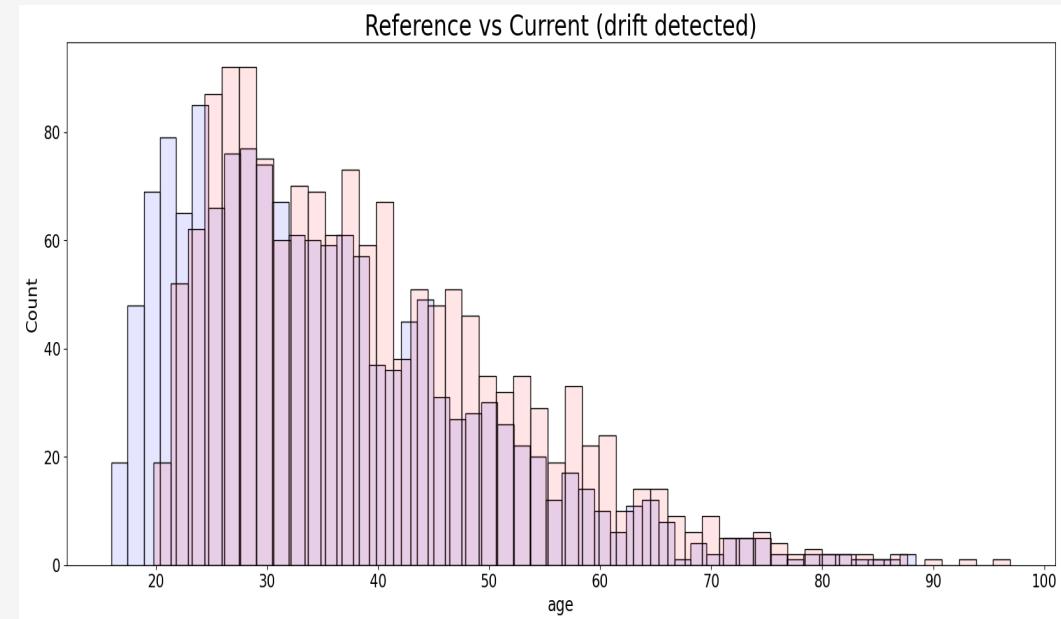
How to detect drift automatically?

- We are not great at detecting drift by eye
- Even if we were, whose job should this be?
- A statistical test compares distributions
- Requests from production are compared with the reference dataset we used for training
- unfortunately there is not one suitable test
- some only fit well for small (< 1000) datasets
- some can work not only on numeric data but also on categorical data
- different tests result in different kind of scores
- scores, like p-values often are not very intuitive

p-values for our two potential drifts using two-sample Kolmogorov-Smirnov (K-S) tests



p-value = 75%, no drift detected



p-value < 0.1%, drift detected

Counter-Measures for drift

- Train new version of the model
 - Record (and label) new data
 - Create new features
 - Change (or fix) model architecture and re-train
- Quick action
 - Recalibrate pre/post processing of model
 - Adjust thresholds for application
 - Exclude certain areas
- Very fast action: fallback
 - Manual Classification
 - Heuristics / Baseline

Drift Detection in Images

- reduce single low level, e.g.
 - structure index
 - mean or std of basic feature
- dimensionality reduction to multivariate data
 - p-values for each feature aggregated
- drift detection via distilled model
 - similar to adversarial attack detection

<https://towardsdatascience.com/detecting-semantic-drift-within-image-data-6a59a0e768c6>

https://docs.seldon.io/projects/alibi-detect/en/stable/examples/cd_ks_cifar10.html

https://docs.seldon.io/projects/alibi-detect/en/stable/examples/cd_distillation_cifar10.html

What can also happen when you go into production

- Need for explanation
 - people sue you over a prediction
 - CEO wants to know why bad score for spouse
 - contradicts prevention of hacking
- Unwanted bias
 - gender or age are common
 - investigative journalism a threat
 - people might again sue

Summary

- a model that is not in production is useless
- getting a descent model is already hard, but...
- bringing a model into production and keeping it there is a whole different story
- machine learning systems typically have to be regularly retrained and maintained
- practitioners need statistical skills

Thanks a lot

Resilient Machine Learning

Stay in Contact

<https://www.linkedin.com/in/oliver-zeigermann-34989773/>

oliver@zeigermann.de

Twitter: @DJCordhose

