Resilient Machine Learning

https://dsco.usfdatainstitute.org, March 2023, San Francisco

Oliver Zeigermann

Slides: https://bit.ly/dsco-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?

Please be as open as possible

Questions and comments are welcome ***at any time***

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)

Please be as open as possible

Questions and comments are welcome ***at any time***

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

Please be as open as possible

Questions and comments are welcome ***at any time***

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

Please be as open as possible

Questions and comments are welcome ***at any time***

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

Please be as open as possible

Questions and comments are welcome ***at any time***

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

Please be as open as possible

Questions and comments are welcome ***at any time***

Gauge by show of hands

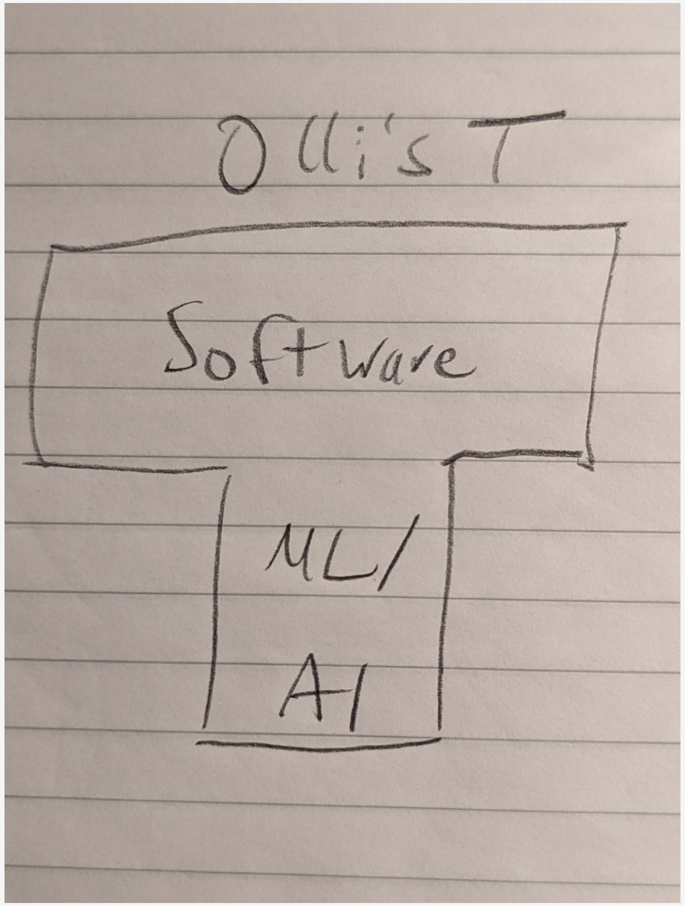
What do you think it most important in a machine learning project?

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

Questions and comments are welcome ***at any time***

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

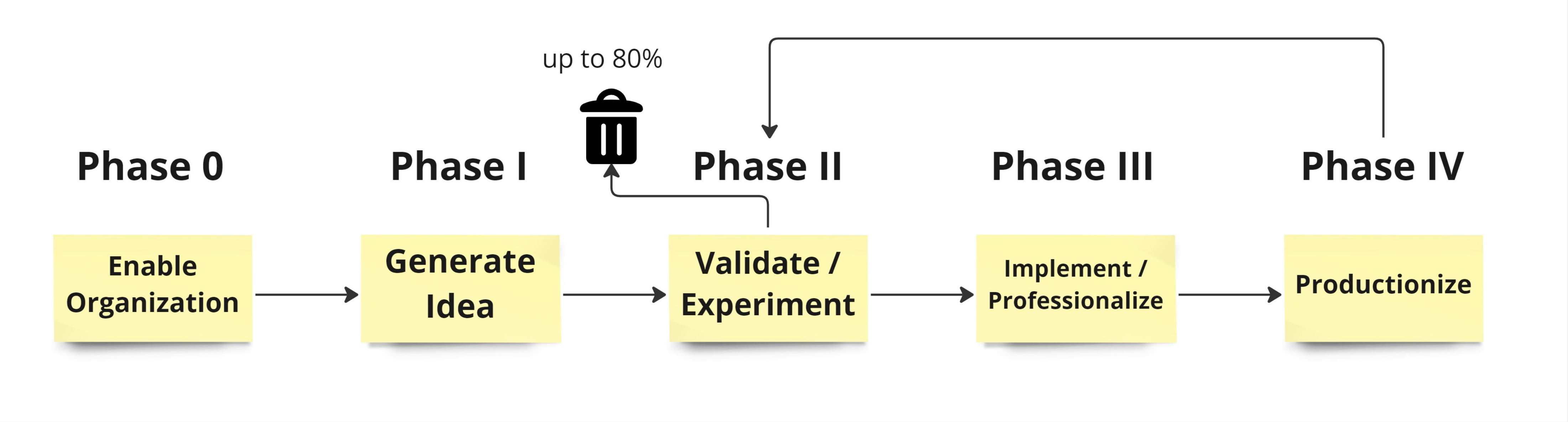
ML comes with a lot of uncertainty

- 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

Machine Learning Projects can be structured in phases



Agenda

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

Without deployment there is no benefit

Without deployment there is no benefit

- in academic life, often only the score (accuracy) of the model counts

Without deployment there is no benefit

- in academic life, often only the score (accuracy) of the model counts
- this approach has become widespread in the field of data science

Without deployment there is no benefit

- in academic life, often only the score (accuracy) of the model counts
- this approach has become widespread in the field of data science
- practice, however, is not a Kaggle Competition

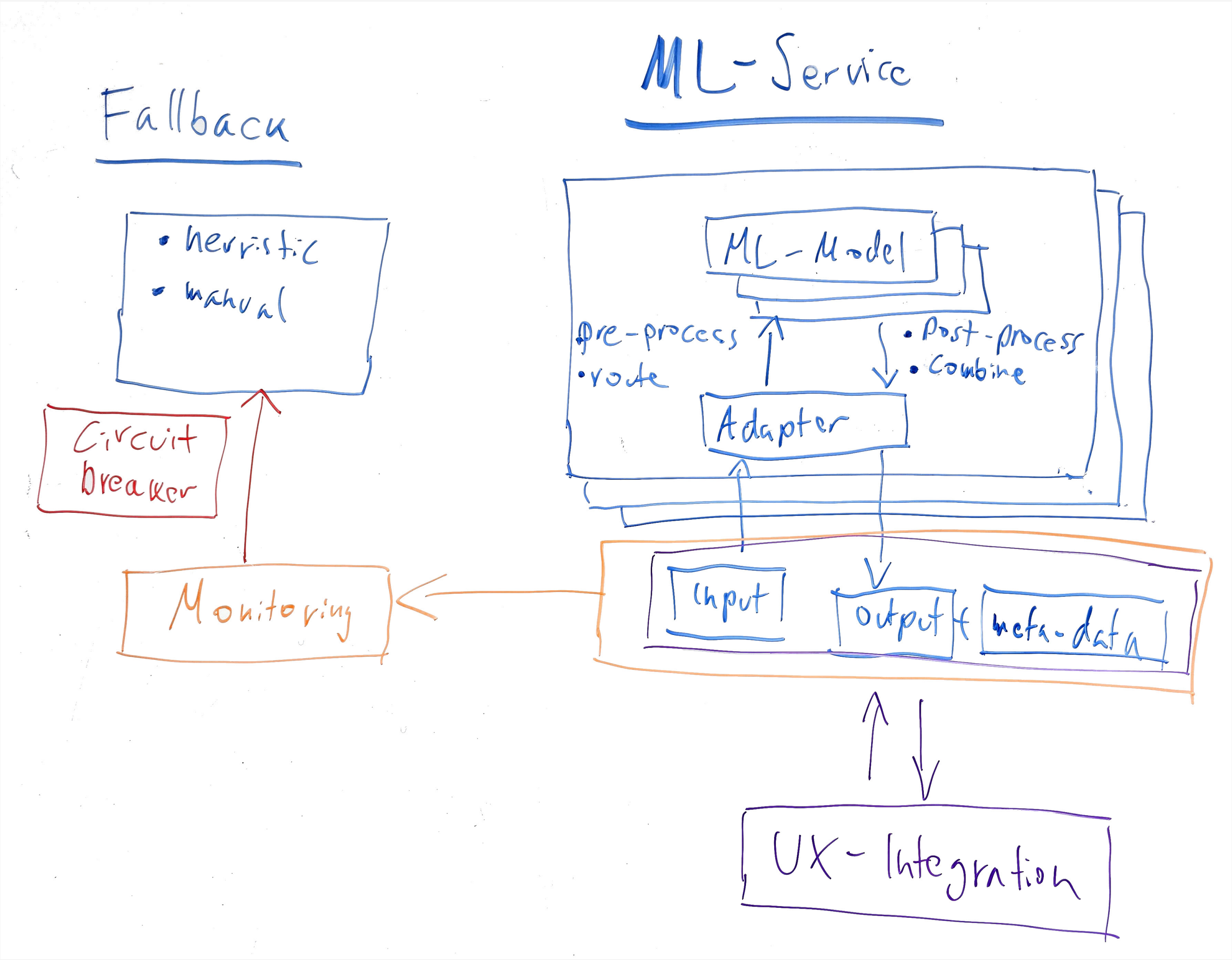
Without deployment there is no benefit

- in academic life, often only the score (accuracy) of the model counts
- this approach has become widespread in the field of data science
- practice, however, is not a Kaggle Competition
- in-sample Evaluation is typically not enough for real life applications

Without deployment there is no benefit

- in academic life, often only the score (accuracy) of the model counts
- this approach has become widespread in the field of data science
- practice, however, is not a Kaggle Competition
- in-sample Evaluation is typically not enough for real life applications
- out-of-sample evaluation typically only possible in production

You don't just deploy the model



Typical requirements for Machine Learning Projects

Typical requirements for Machine Learning Projects

1. ***Reasonable Accuracy*** - should have 80% or more cases correct

Typical requirements for Machine Learning Projects

1. ***Reasonable Accuracy*** - should have 80% or more cases correct
2. ***Explainability*** - we need to explain how we come to the conclusion

Typical requirements for Machine Learning Projects

1. ***Reasonable Accuracy*** - should have 80% or more cases correct
2. ***Explainability*** - we need to explain how we come to the conclusion
3. ***Stability*** - a new version of the software should not behave totally different

Typical requirements for Machine Learning Projects

1. ***Reasonable Accuracy*** - should have 80% or more cases correct
2. ***Explainability*** - we need to explain how we come to the conclusion
3. ***Stability*** - a new version of the software should not behave totally different
4. ***Resistance Against Manipulation*** - it should be hard to find out how the software works to resist manipulation

Typical requirements for Machine Learning Projects

1. ***Reasonable Accuracy*** - should have 80% or more cases correct
2. ***Explainability*** - we need to explain how we come to the conclusion
3. ***Stability*** - a new version of the software should not behave totally different
4. ***Resistance Against Manipulation*** - it should be hard to find out how the software works to resist manipulation
5. ***Robustness against drift*** - the world changes, our model should follow

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=MDJmNzVkMjY=

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

Attempted Manipulation / Adversarial Attacks

Certain inputs may intentionally cause a grossly incorrect prediction to be made

Attempted Manipulation / Adversarial Attacks

Certain inputs may intentionally cause a grossly incorrect prediction to be made

- can be used to systematically manipulate the output of the system

Attempted Manipulation / Adversarial Attacks

Certain inputs may intentionally cause a grossly incorrect prediction to be made

- can be used to systematically manipulate the output of the system
- how sensitive a system is to such an attempt is determined by the inner complexity of such a system

Attempted Manipulation / Adversarial Attacks

Certain inputs may intentionally cause a grossly incorrect prediction to be made

- can be used to systematically manipulate the output of the system
- how sensitive a system is to such an attempt is determined by the inner complexity of such a system
  - often directly contradicting properties like explainability

Model stability and adversarial vulnerability

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

If so

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

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

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

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

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

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

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.

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

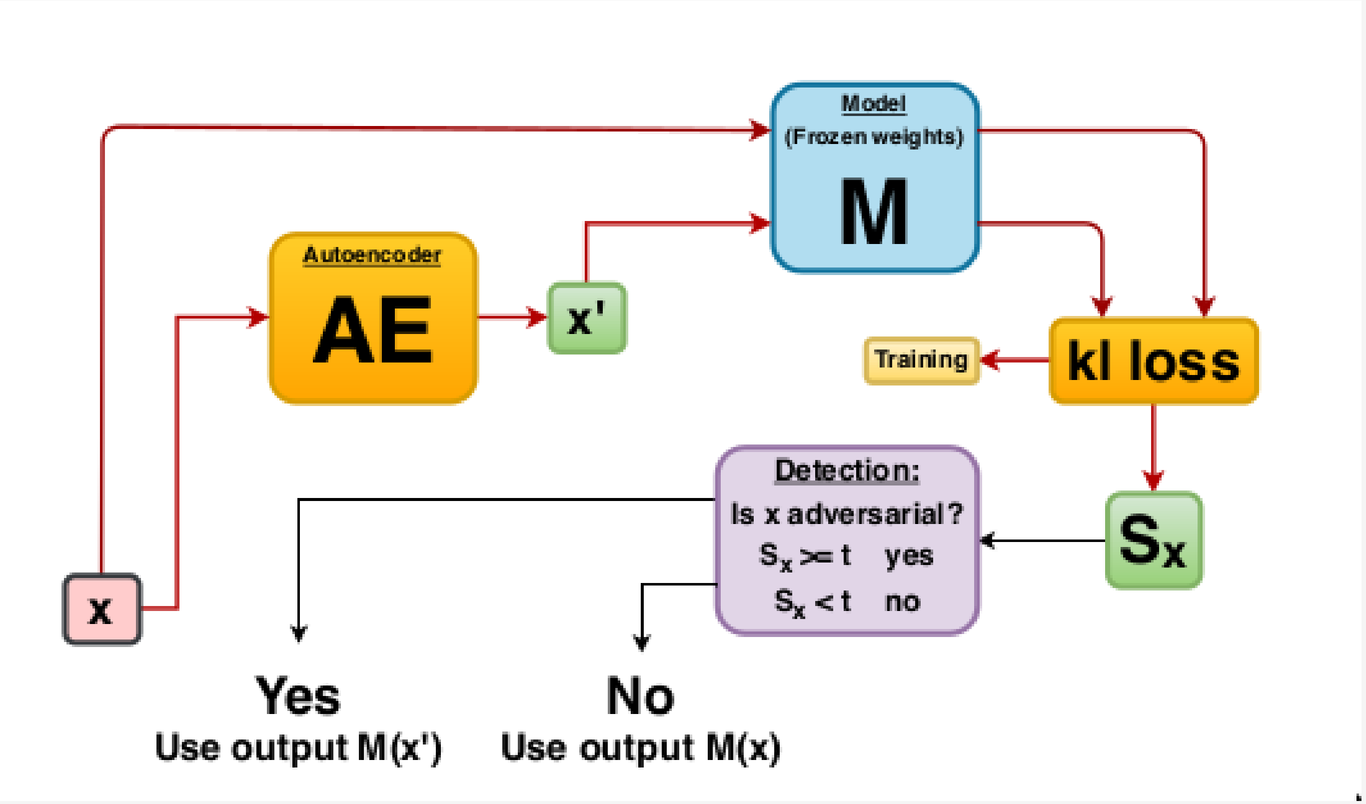
Adversarial AE detector

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
https://arxiv.org/abs/2002.09364

Adversarial AE detector

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

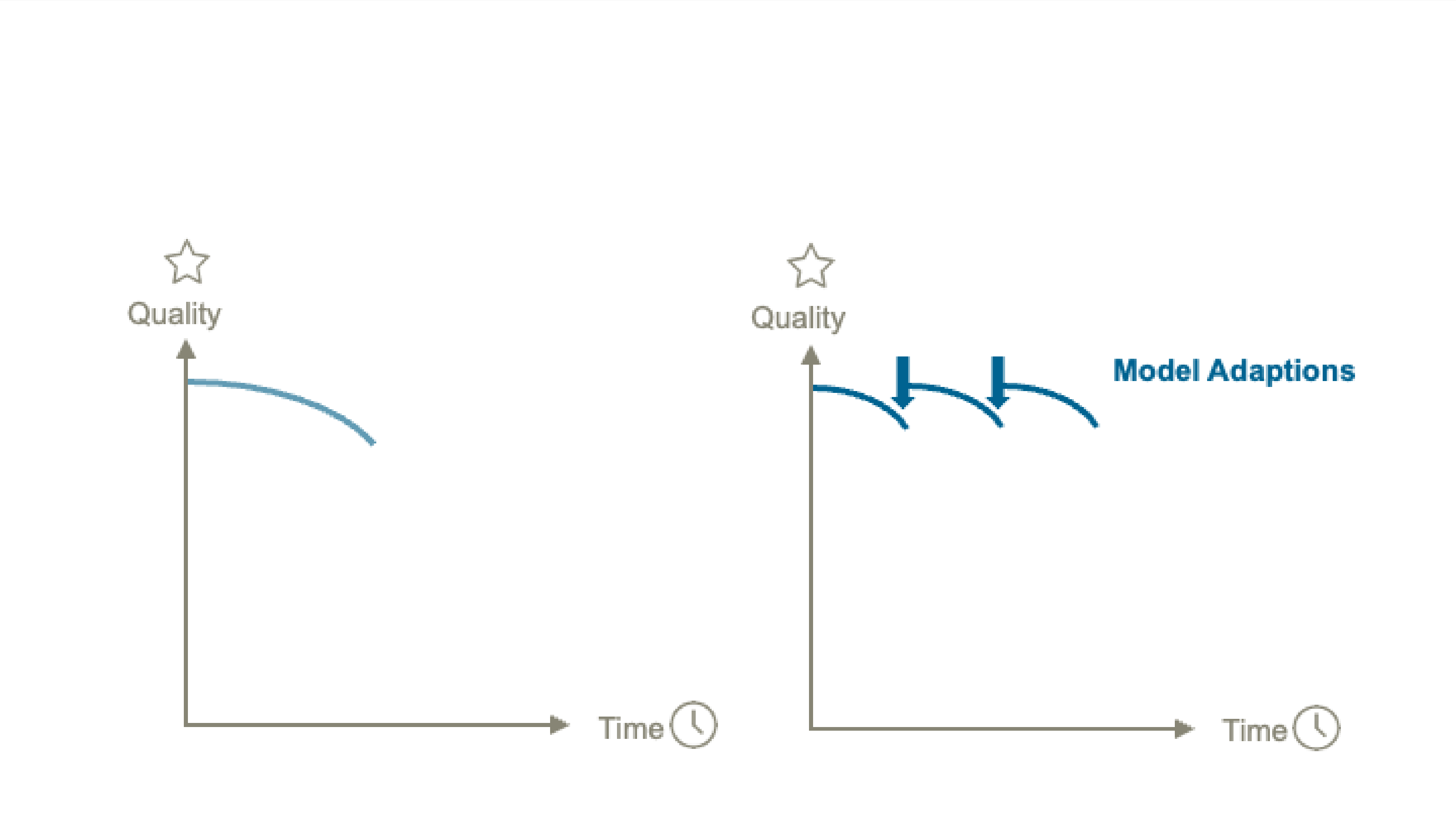


https://docs.seldon.io/projects/alibi-detect/en/stable/ad/methods/adversarialae.html
https://github.com/SeldonIO/alibi-detect#adversarial-detection
https://arxiv.org/abs/2002.09364

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?

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

- Check how the model behaves on newer data

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

- Check how the model behaves on newer data
  - How quickly does performance degrade?

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

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

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

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

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

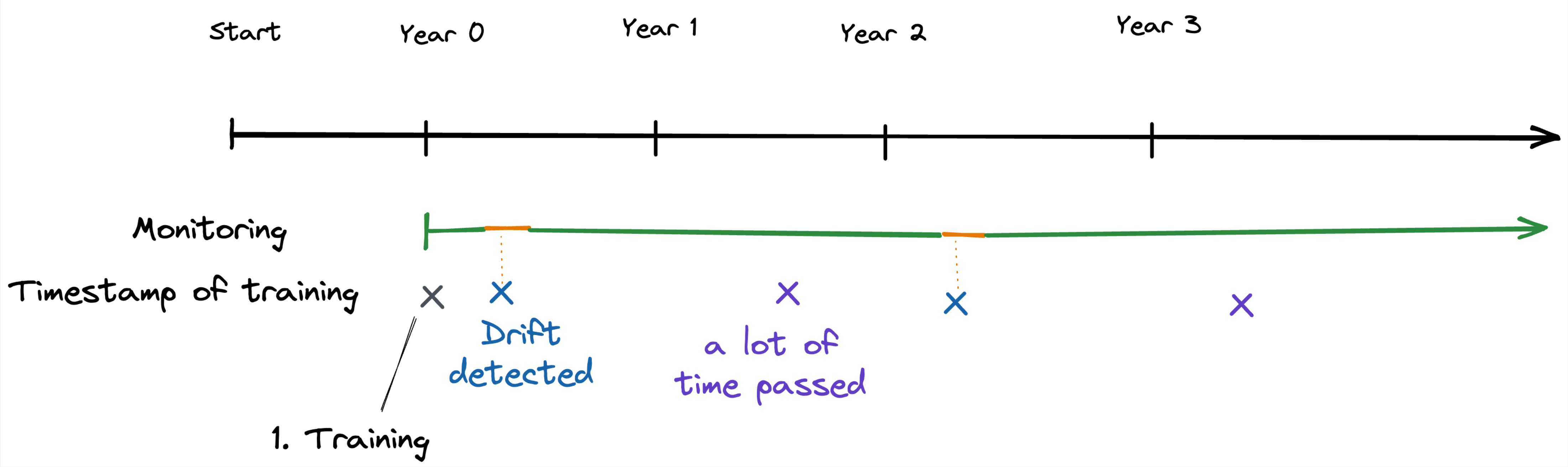
How do you know you need a new model in production?

- 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

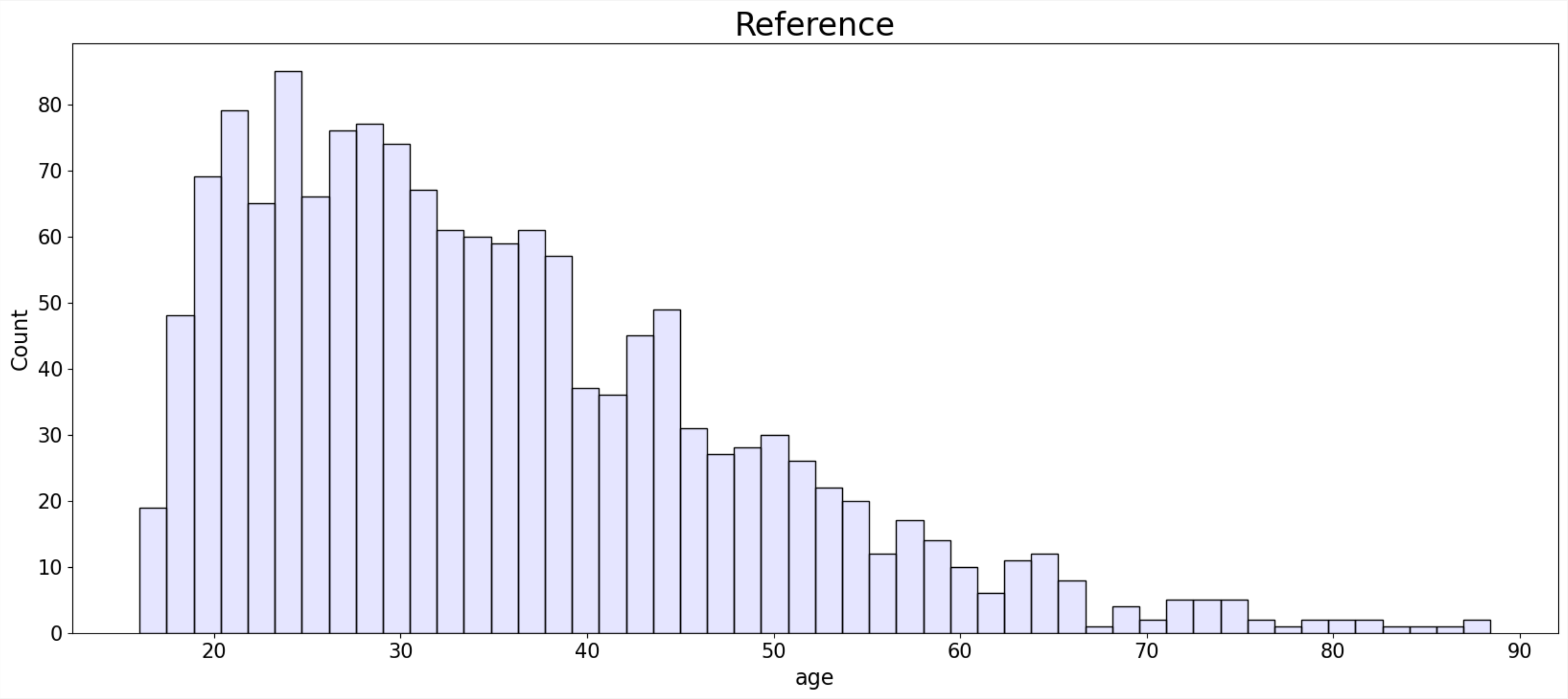
How do you know you need a new model in production?

- 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?

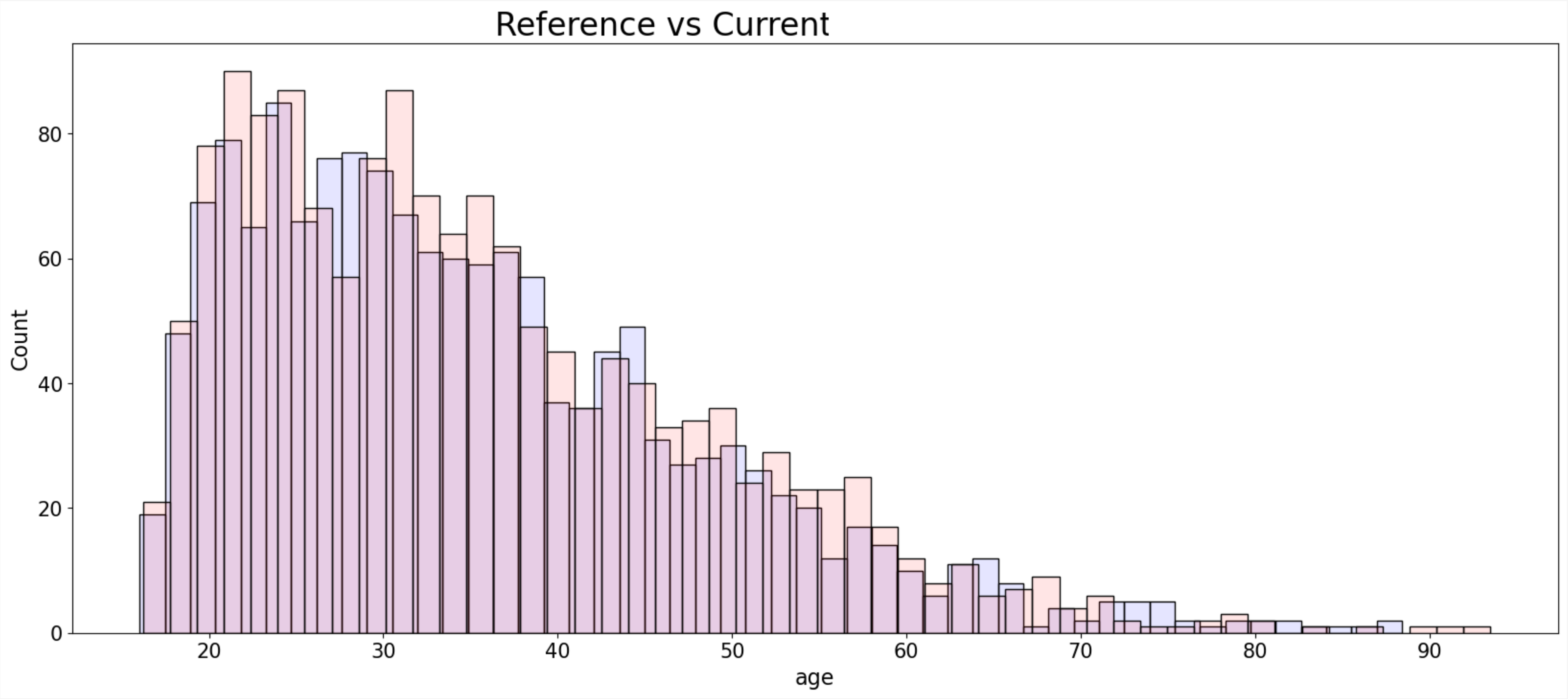


Reference Distribution

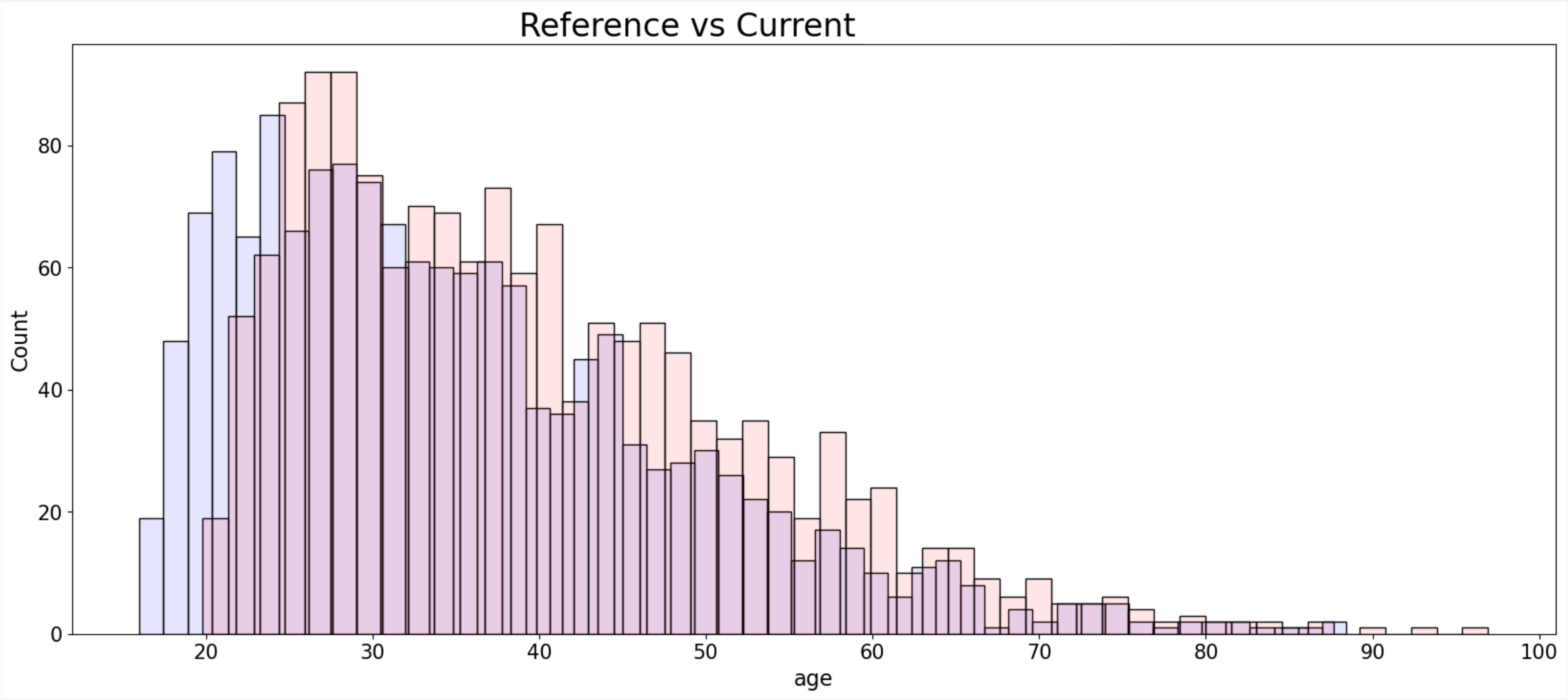


Example: Age of people seeking for insurance

Does it drift?



And this?



Reference vs Current
Count
age
0
20
40
60
80
20
30
40
50
60
70
80
90
100

How to detect drift automatically?

How to detect drift automatically?

- We are not great at detecting drift by eye

How to detect drift automatically?

- We are not great at detecting drift by eye
- Even if we were, whose job should this be?

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

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

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

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

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

How to detect drift automatically?

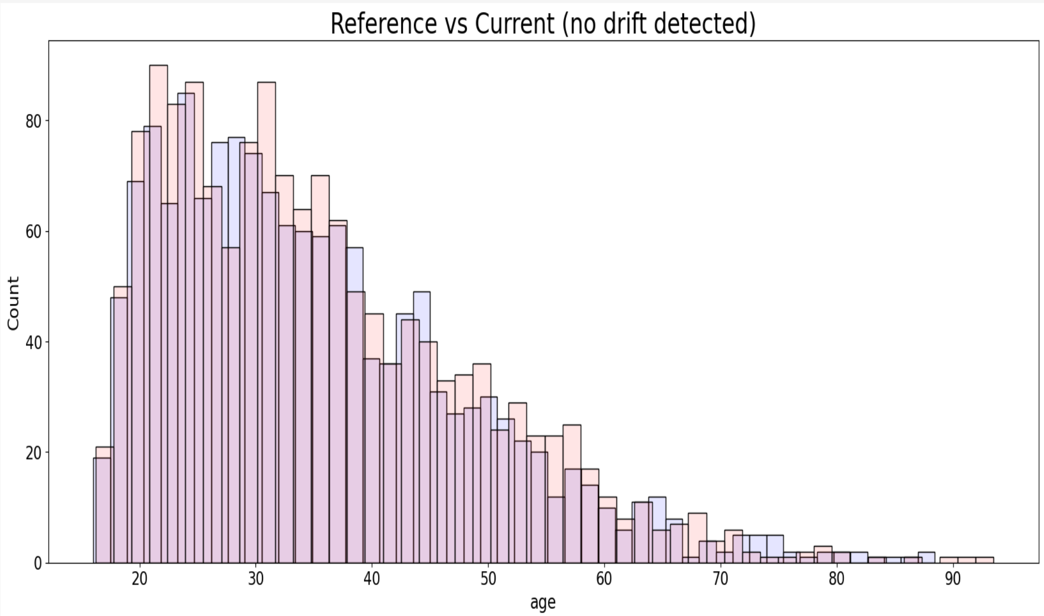
- We are not great at detecting drift by eye
- Even if we were, whose job should this be?
- A statistical test compares distributions
- Requests from production are compared with the reference dataset we used for training
- unfortunately there is not one suitable test
- some only fit well for small (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

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

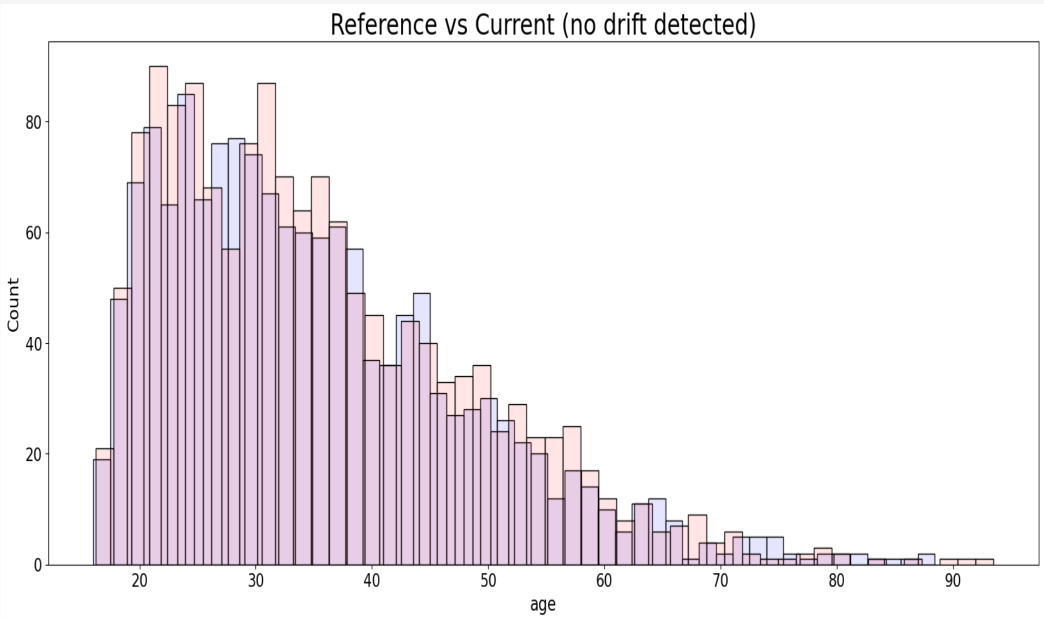
p-values for our two potential drifts using two-sample Kolmogorov-Smirnov (K-S) tests

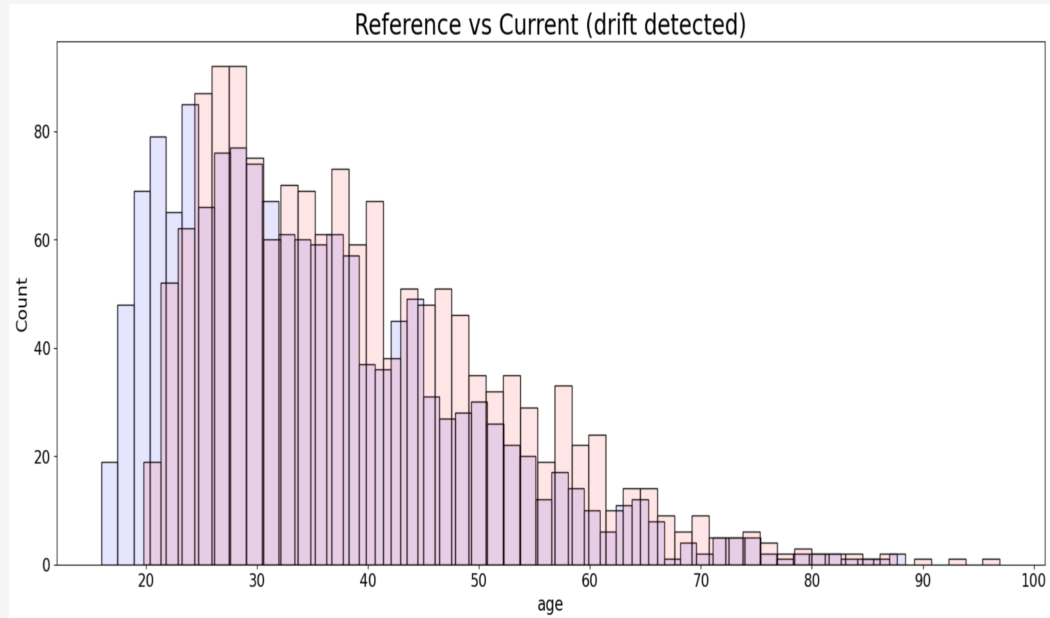
p-values for our two potential drifts using two-sample Kolmogorov-Smirnov (K-S) tests



p-value = 75%, no drift detected

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

Counter-Measures for drift

- Train new version of the model

Counter-Measures for drift

- Train new version of the model
  - Record (and label) new data

Counter-Measures for drift

- Train new version of the model
  - Record (and label) new data
  - Create new features

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

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

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

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

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

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

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

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

What can also happen when you go into production

What can also happen when you go into production

- Need for explanation

What can also happen when you go into production

- Need for explanation
  - people sue you over a prediction

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

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

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

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

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

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

Drift Detection in Images

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

Drift Detection in Images

- reduce single low level, e.g.

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

Drift Detection in Images

- reduce single low level, e.g.
  - structure index

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

Drift Detection in Images

- reduce single low level, e.g.
  - structure index
  - mean or std of basic feature

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

Drift Detection in Images

- reduce single low level, e.g.
  - structure index
  - mean or std of basic feature
- dimensionality reduction to multivariate data

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

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

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

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

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

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

Summary

Summary

- a model that is not in production is useless

Summary

- a model that is not in production is useless
- getting a descent model is already hard, but...

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

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 projects can be thought of in phases

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 projects can be thought of in phases
- in production, special challenges arise in the area of monitoring

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 projects can be thought of in phases
- in production, special challenges arise in the area of monitoring
- machine learning systems typically have to be regularly retrained and maintained

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 projects can be thought of in phases
- in production, special challenges arise in the area of monitoring
- 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