GOAD🐐

Goal Oriented Analysis of Data

Raoul Grouls, 24 maart 2025

https://github.com/raoulg/goad_toolkit

https://github.com/raoulg/codestyle

Topic	Explore	Consolidate	Cooperate	Deploy
never hardcode	💡	🥇	🥇	🥇
Prefer pathlib.Path over os.path	💡	🥇	🥇	🥇
pyproject.toml for dependencies	💡	🥇	🥇	🥇
organize your folders	💡	🥇	🥇	🥇
Add a README	💡	🥇	🥇	🥇
isolate your settings	💡	💡	🥇	🥇
classes and inheritance	💡	💡	🥇	🥇
make your code abstract enough	💡	💡	🥇	🥇
Git	💡	💡	🥇	🥇
Use formatters and linting	🐌	💡	🥇	🥇
use logging	🐌	💡	🥇	🥇
Use typehinting	🐌	💡	🥇	🥇
Make a proper module	🐌	💡	🥇	🥇
Encapsulation, SRP	🐌	💡	🥇	🥇
testing	🐌	💡	🥇	🥇
Open-Closed Principle	🐌	💡	💡	🥇
Makefiles or shell scripts	🐌	💡	💡	🥇
Abstract classes (ABC, Protocol)	🐌	🐌	💡	🥇

Single Responsibility Principle

Every class should have one, and only one, reason to change.

In GOAD🐐:

- `ZScaler` only standardizes values.
- `PlotSettings` only holds plot config.
- `LinePlot` only draws a line.

Each class has one job. Why it matters:

- Easier to read
- Easier to debug
- Easier to extend without breaking stuff

Open-Closed Principle

Software entities should be open for extension, but closed for modification.

In GOAD🐐:

- You can add a new transform by subclassing `TransformBase`
- You can register a new distribution with `DistributionRegistry`
- You can build your own `BasePlot` component

You can add new functionality without rewriting core code!

Singleton

When a class has one instance, accessible globally

```python
@dataclass
class DistributionRegistry:
    """Registry for statistical distributions with metadata."""

    _instance = None

    def __new__(cls):
        """Ensure only one instance of DistributionRegistry exists."""
        if cls._instance is None:
            cls._instance = super().__new__(cls)
            cls._instance.initialized = False
        return cls._instance

    def __init__(self):
        """Initialize the registry with common distributions."""
        if not self.initialized:
            self.distributions: Dict[str, Distribution] = {}
            self.register_distribution(
                "norm", stats.norm, is_discrete=False, num_params=2
            )
            self.register_distribution(
                "uniform", stats.uniform, is_discrete=False, num_params=2
            )
            self.register_distribution(
                "lognorm", stats.lognorm, is_discrete=False, num_params=3
            )
            self.register_distribution(
                "poisson", stats.poisson, is_discrete=True, num_params=1
            )
            self.register_distribution(
                "exponential", stats.expon, is_discrete=False, num_params=2
            )
            self.register_distribution(
                "skewnorm", stats.skewnorm, is_discrete=False, num_params=3
            )
            self.register_distribution(
                "gamma", stats.gamma, is_discrete=False, num_params=3
            )
            self.register_distribution(
                "weibull", stats.weibull_min, is_discrete=False, num_params=3
            )
            self.initialized = True
```

Machine learning

Machine Learning

The summary

$$f\colon X \to y$$


0.38
3.50
7.26
4.56
6.30
7.86
3.30
0

The map is not the territory

- Models are simplifications
- Models are built on assumptions
- The map might be correct, but the territory can change
- The simplicity of the models shows what is otherwise difficult to see

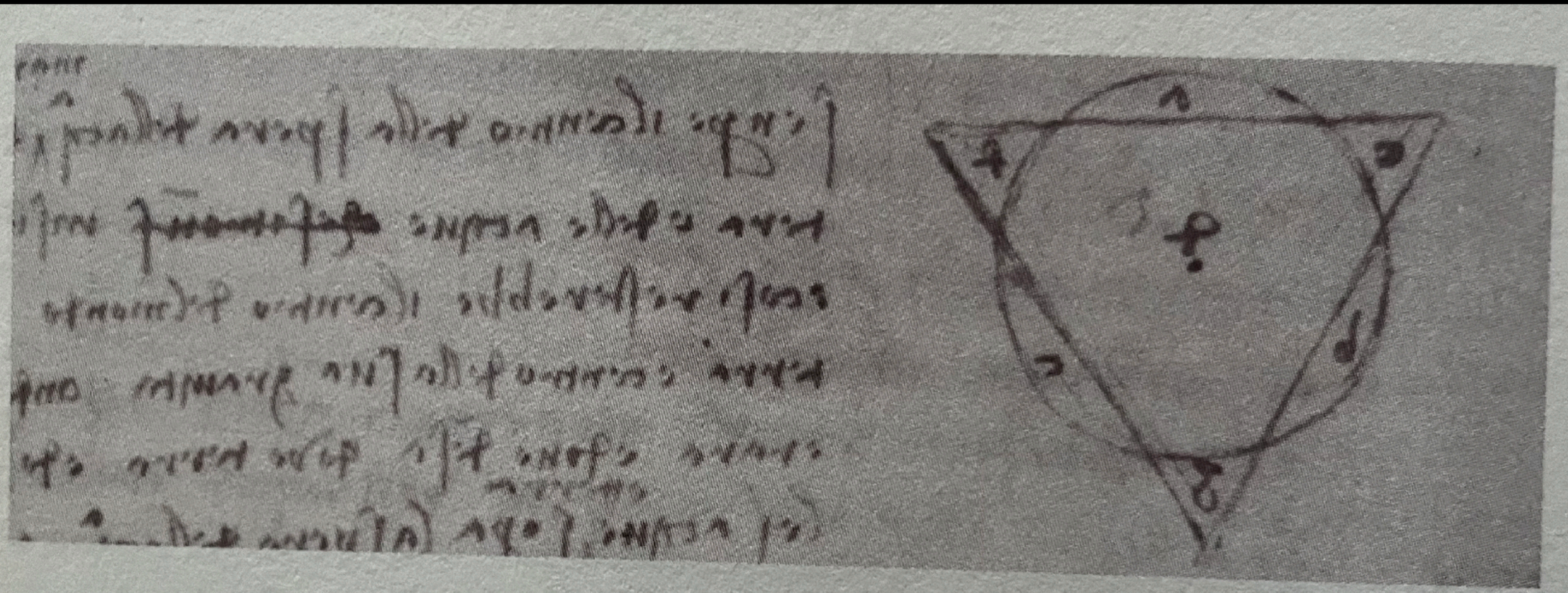
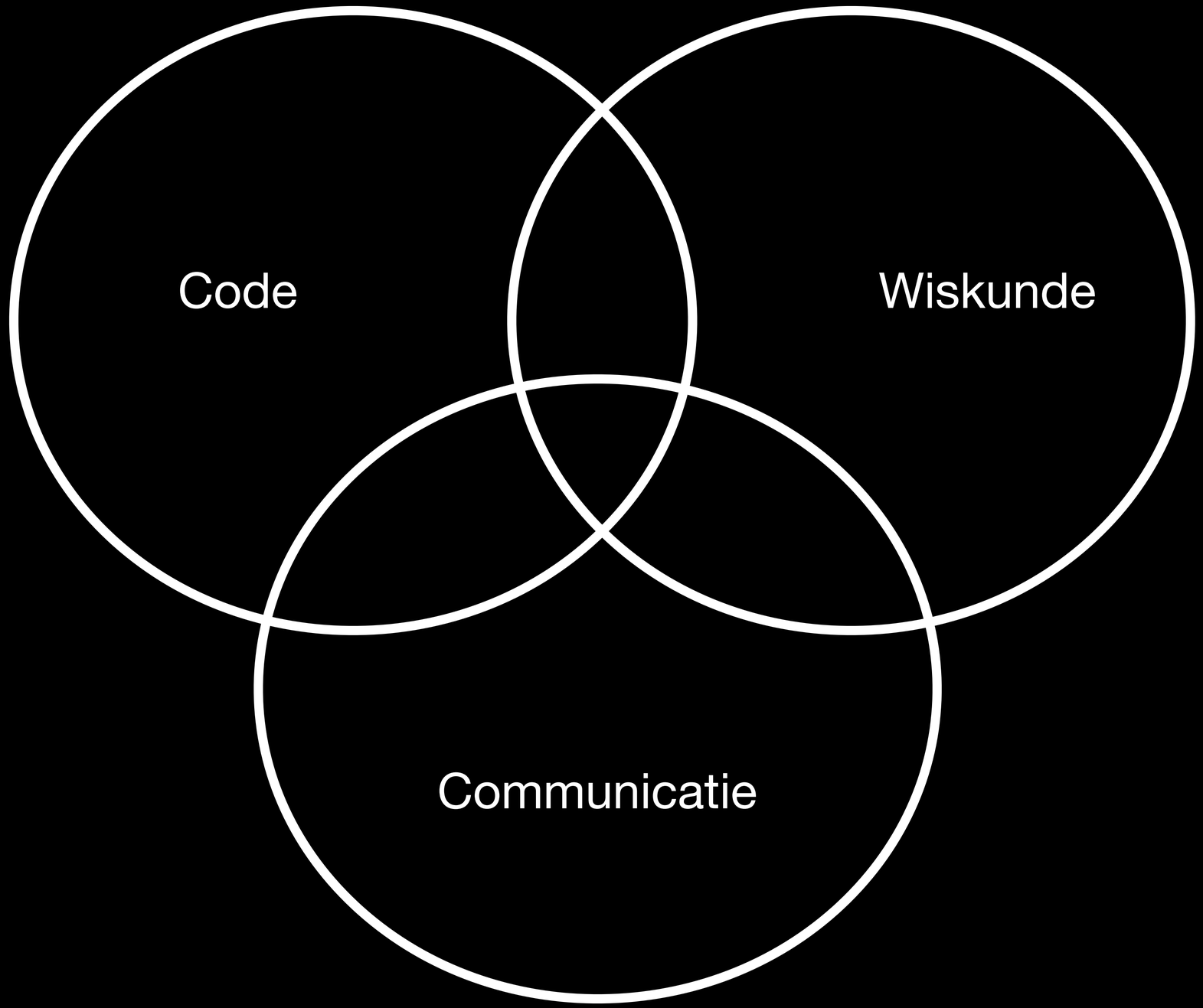
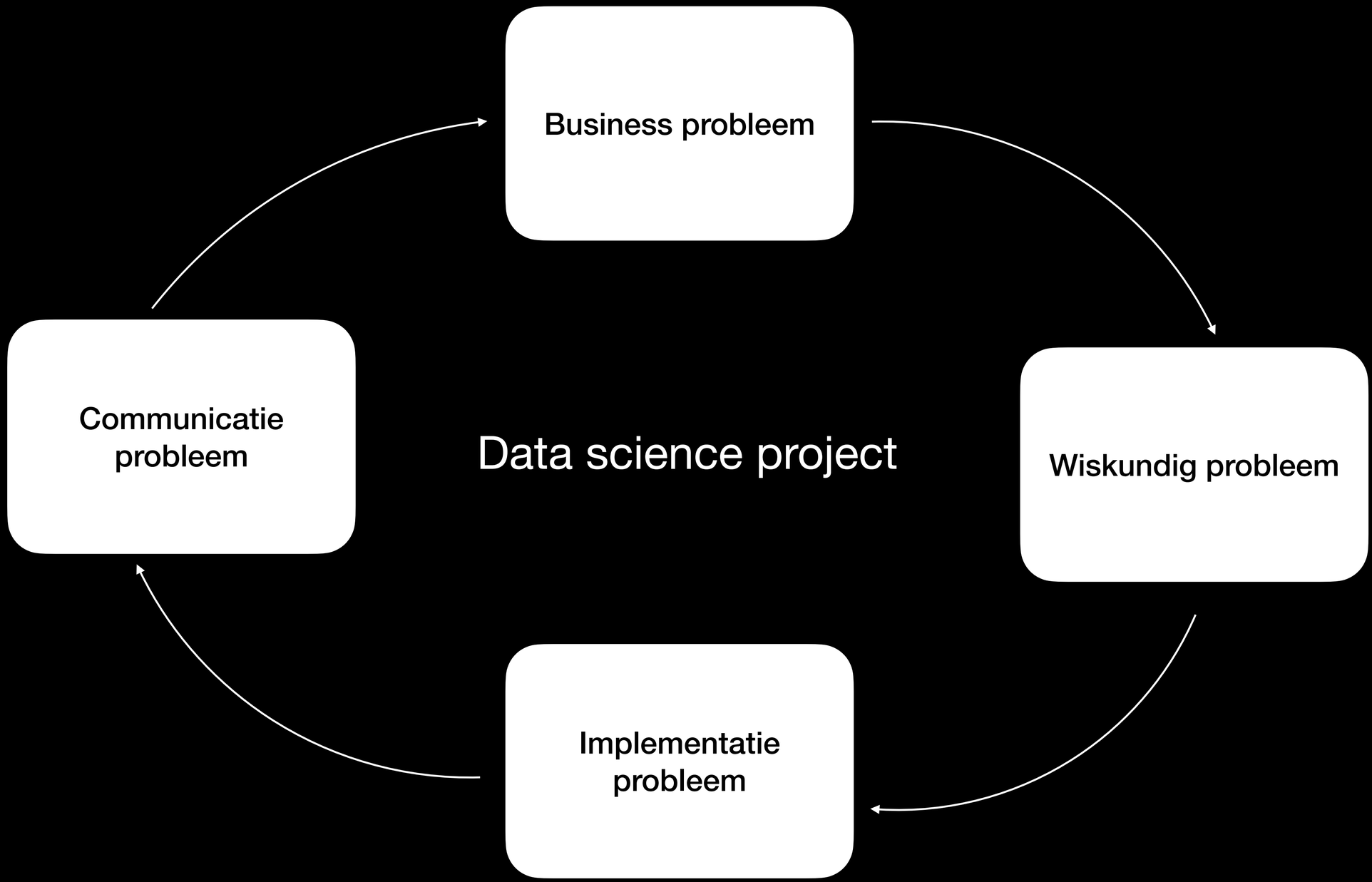


FIG. 2-7. Geometric model of the Earth.
Codex Leicester, folio 35v (detail).


Code
Wiskunde
Communicatie


Business probleem
Wiskundig probleem
Implementatie probleem
Communicatie probleem
Data science project

Common student mistakes

The Slot Machine Syndrome

- Treat modeling like a gambling addict: "Just one more model might hit the jackpot!"
- When metrics disappoint, double down and test even MORE models hoping the metrics will give you that hit of dopamine

The Wild Goose Error Chase

- If that fails, start using ChatGPT which overcomplicates your code even more and sends you off on a wild goose chase for errors

The Analysis-Avoidance Disorder

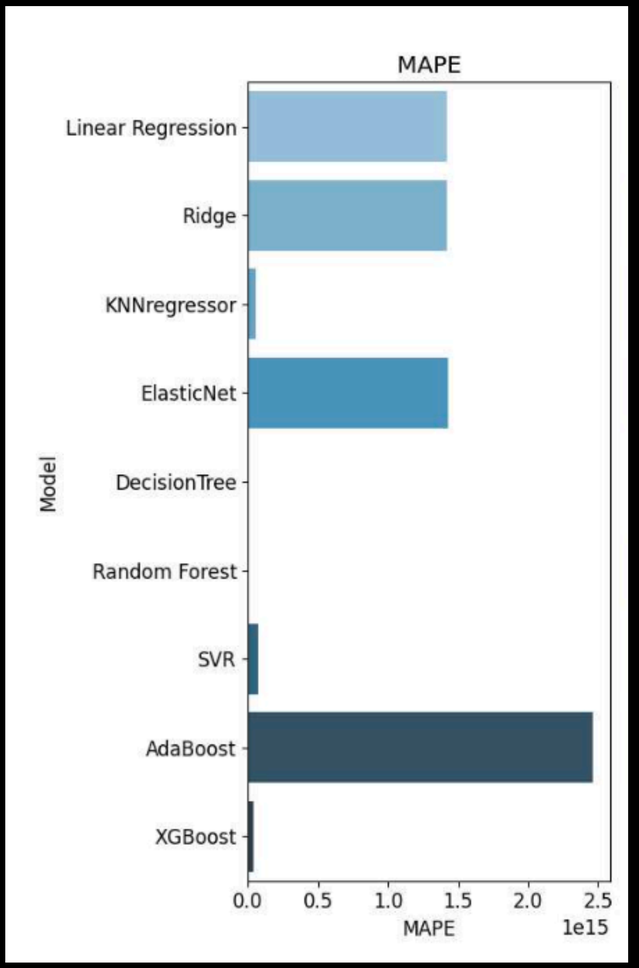
- Hope more computing power will magically solve conceptual problems
- “Unfortunately I ran out of time”

The Sweep-It-Under-The-Rug Conclusion

- Declare your metrics "unfit for this particular problem" when they don't look good, or simply ignore them.
- Have ChatGPT write a conclusion that reads like you are selling a second hand car, describing your model as “robust machine learning”

Do not repeat this 😭

- “After a technical analysis of the model, tuning of the hyperparameters, and focus on the nature of the data, the model achieved an RMSE of 2.413 kW…”
- “These results suggest that the model is a robust and reliable tool for predicting power draw of the engine…”
- “… while MAPE is a useful metric for understanding error in terms of percentage, the extreme high values suggest either that there are extreme outliers in the data or that MAPE is not a well-suited metric for this particular dataset.”

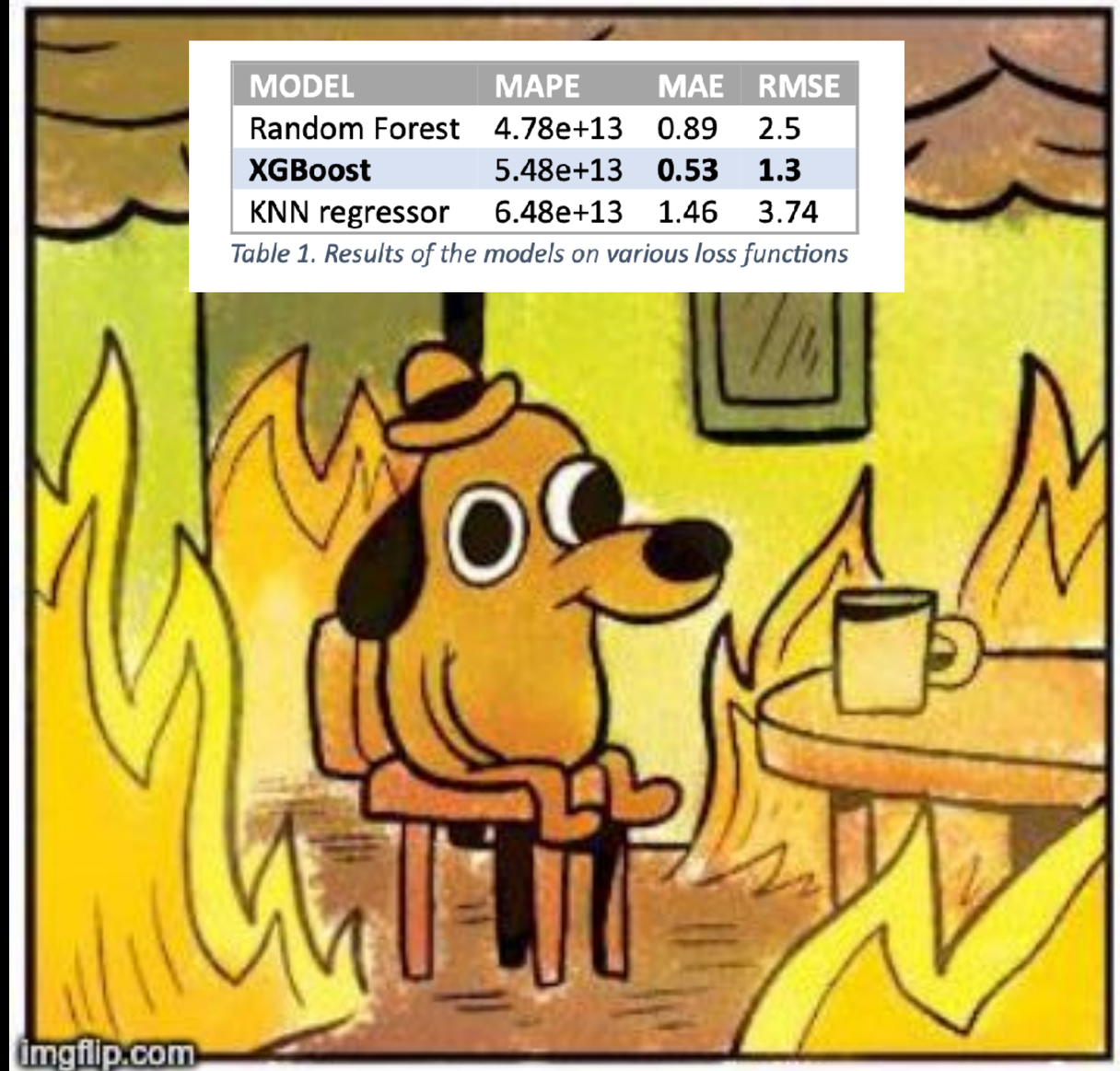


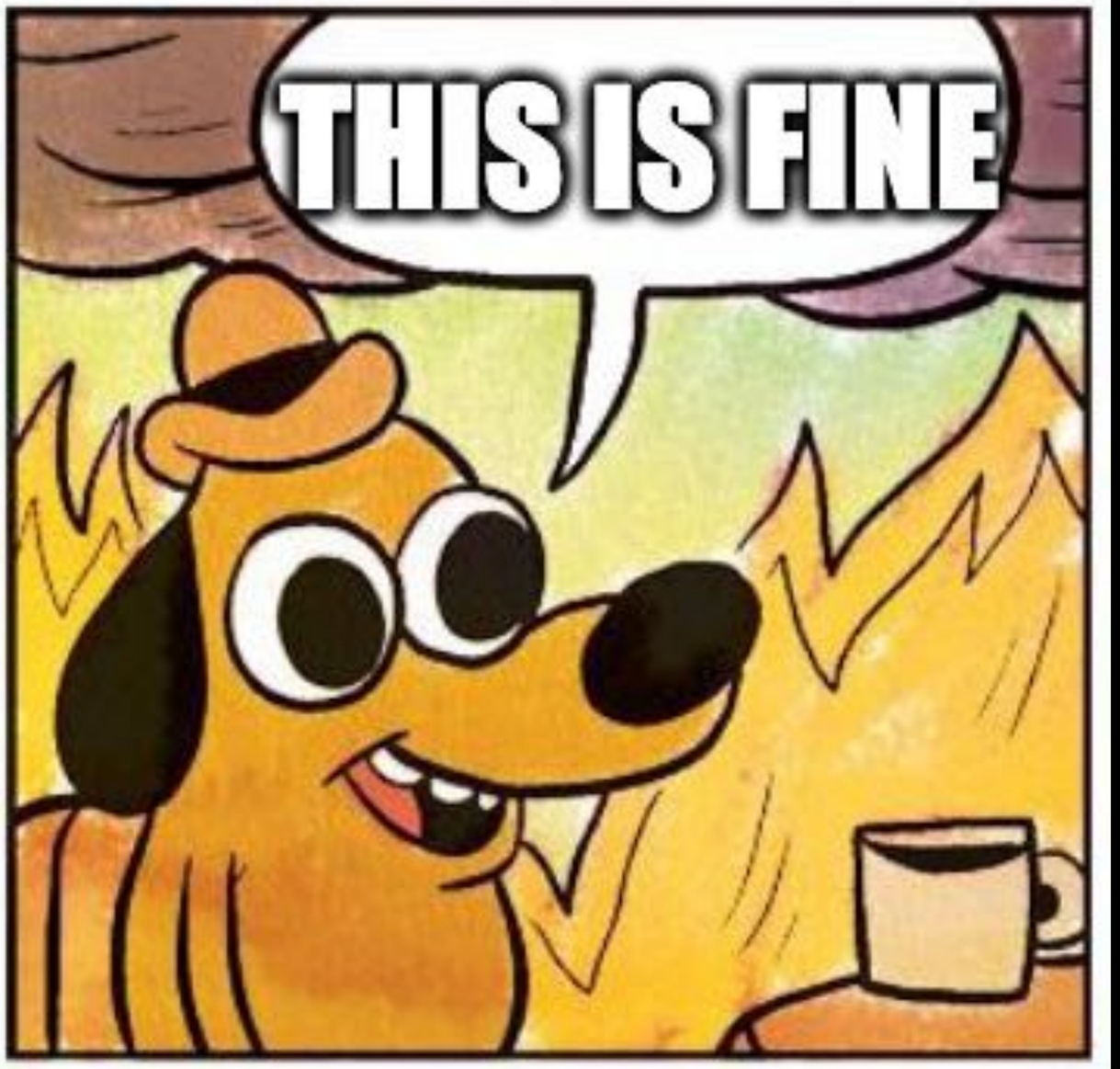
MODEL	MAPE	MAE	RMSE
Random Forest	4.78e+13	0.89	2.5
XGBoost	5.48e+13	**0.53**	**1.3**
KNN regressor	6.48e+13	1.46	3.74

Table 1. Results of the models on various loss functions

MODEL	MAPE	MAE	RMSE
Random Forest	4.78e+13	0.89	2.5
XGBoost	5.48e+13	**0.53**	**1.3**
KNN regressor	6.48e+13	1.46	3.74

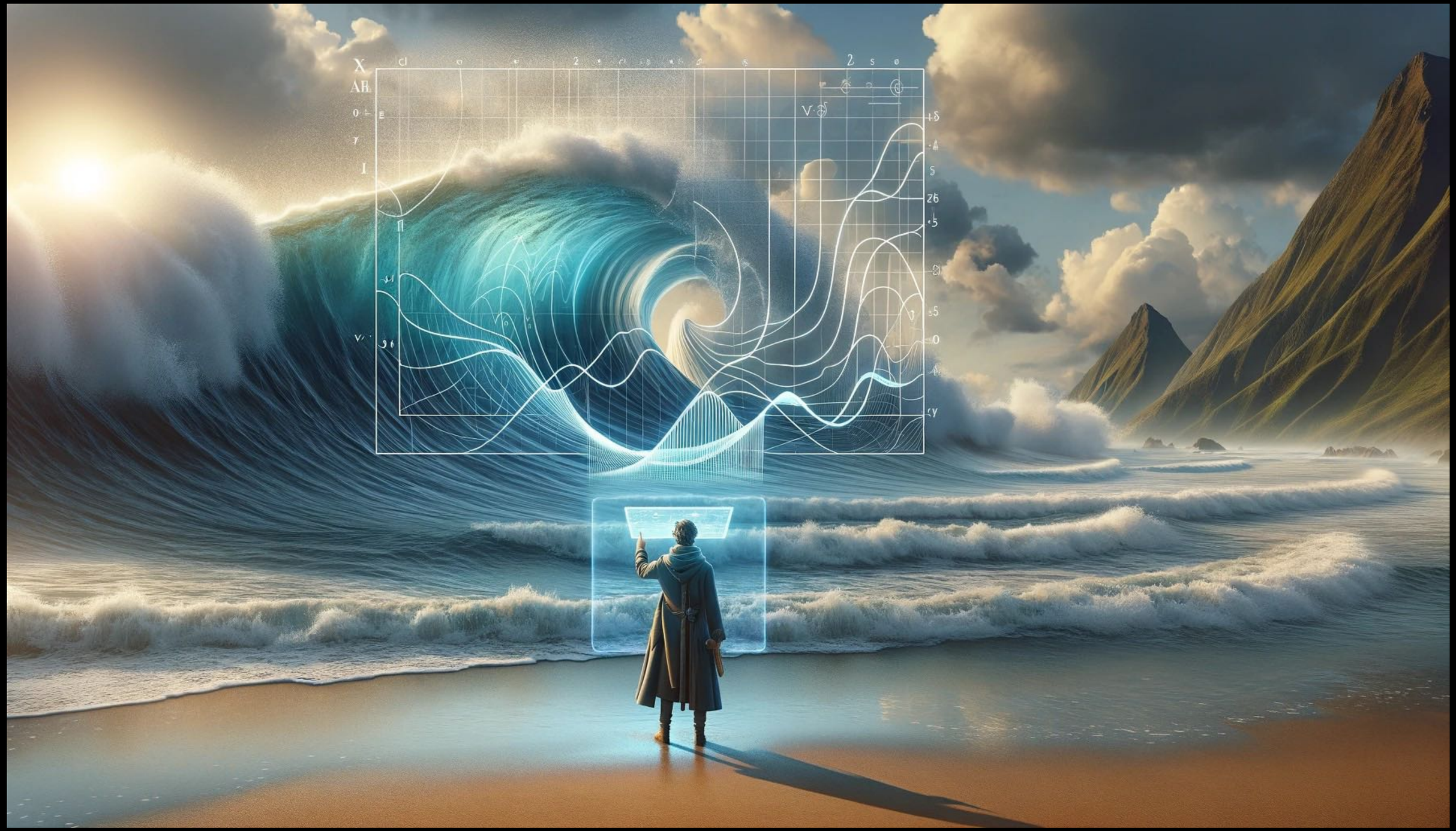
Table 1. Results of the models on various loss functions





What you should do

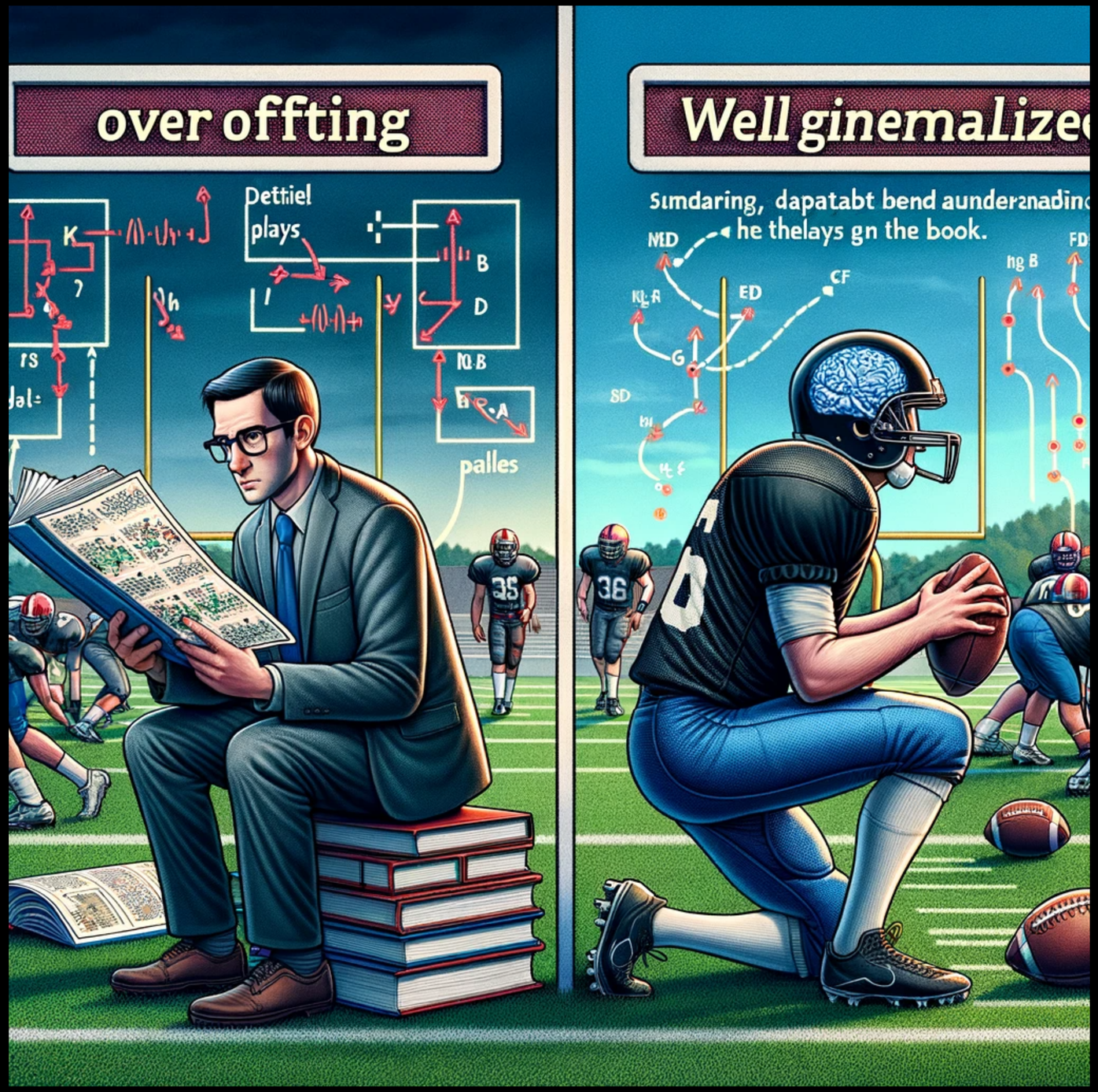
Stop, reflect, and simplify



Overfitting

A model that corresponds too closely to a particular set of data, therefore failing to fit to unseen data.

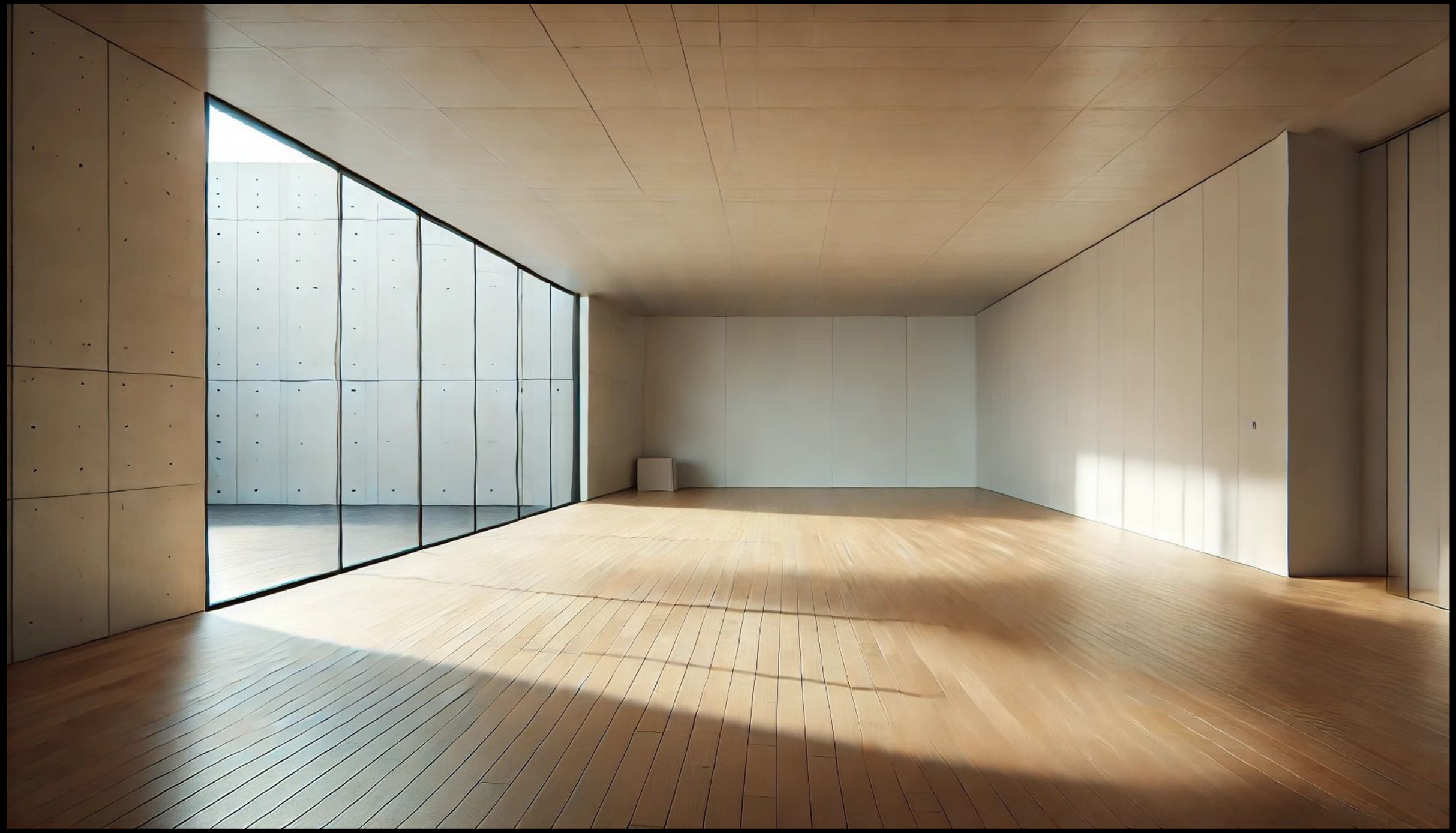
- the error / performance metrics for the training data are very good, but bad on unseen data.
- Overfit models tend to have higher complexity (number of learnable parameters) than is warranted by the data complexity
- Small changes in the data can lead to large changes in the model.

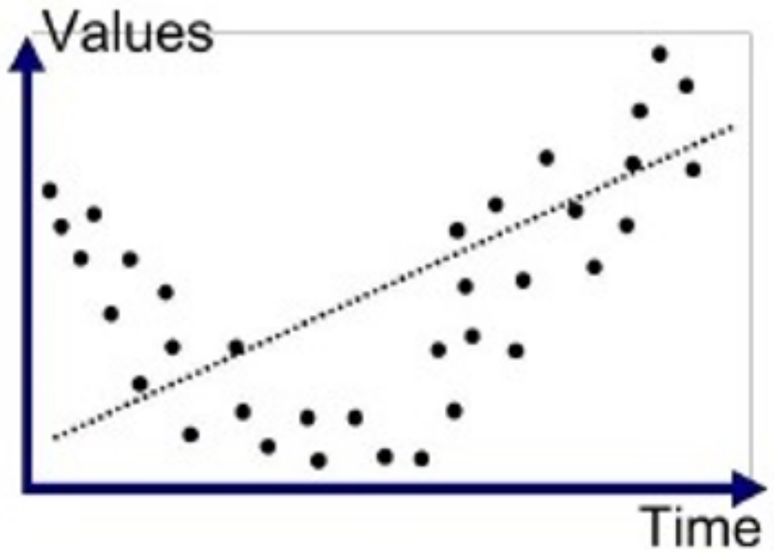


Underfitting

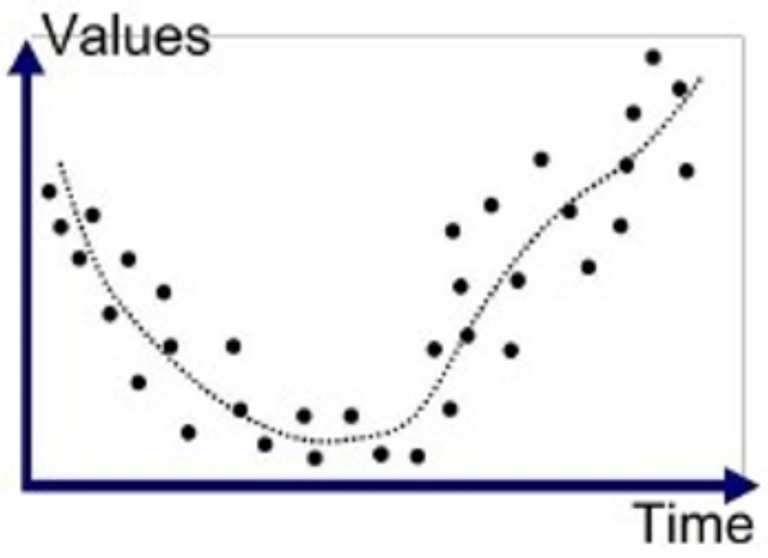
A model that is **too simplistic** to capture the underlying patterns in data, therefore failing to perform well on both training and unseen data.

- The error/performance metrics are poor on both training and test data.
- Underfit models tend to have **lower complexity** (fewer learnable parameters) than is warranted by the data complexity.
- Small changes in the data typically lead to **minimal changes** in the model predictions.

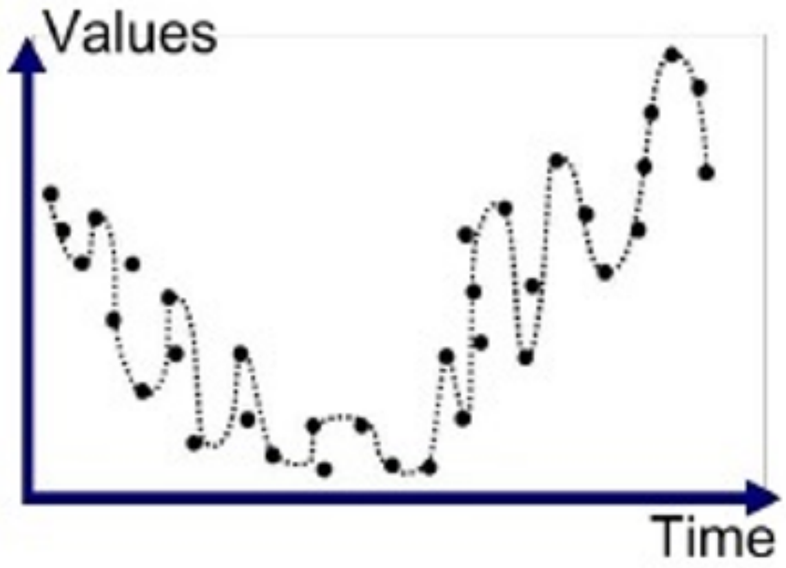




Underfitted



Good Fit/Robust



Overfitted

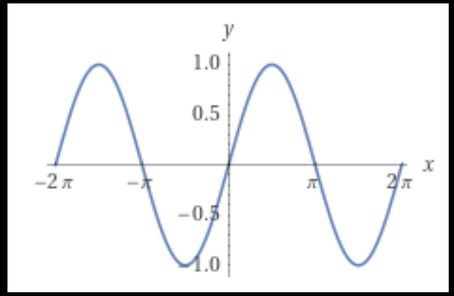
Basic modelling

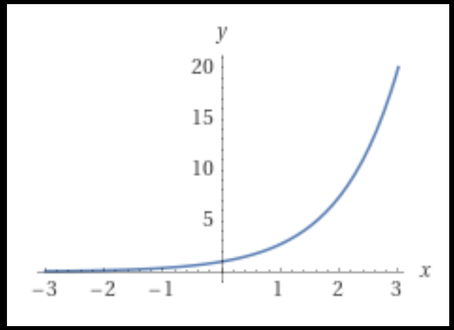
The four horsemen of modelling

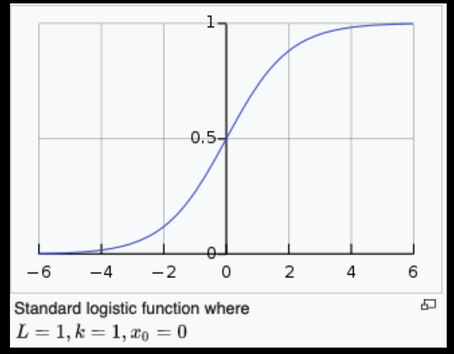
- Linear
- Sine
- Logistic
- Exponential

The four horsemen of modelling

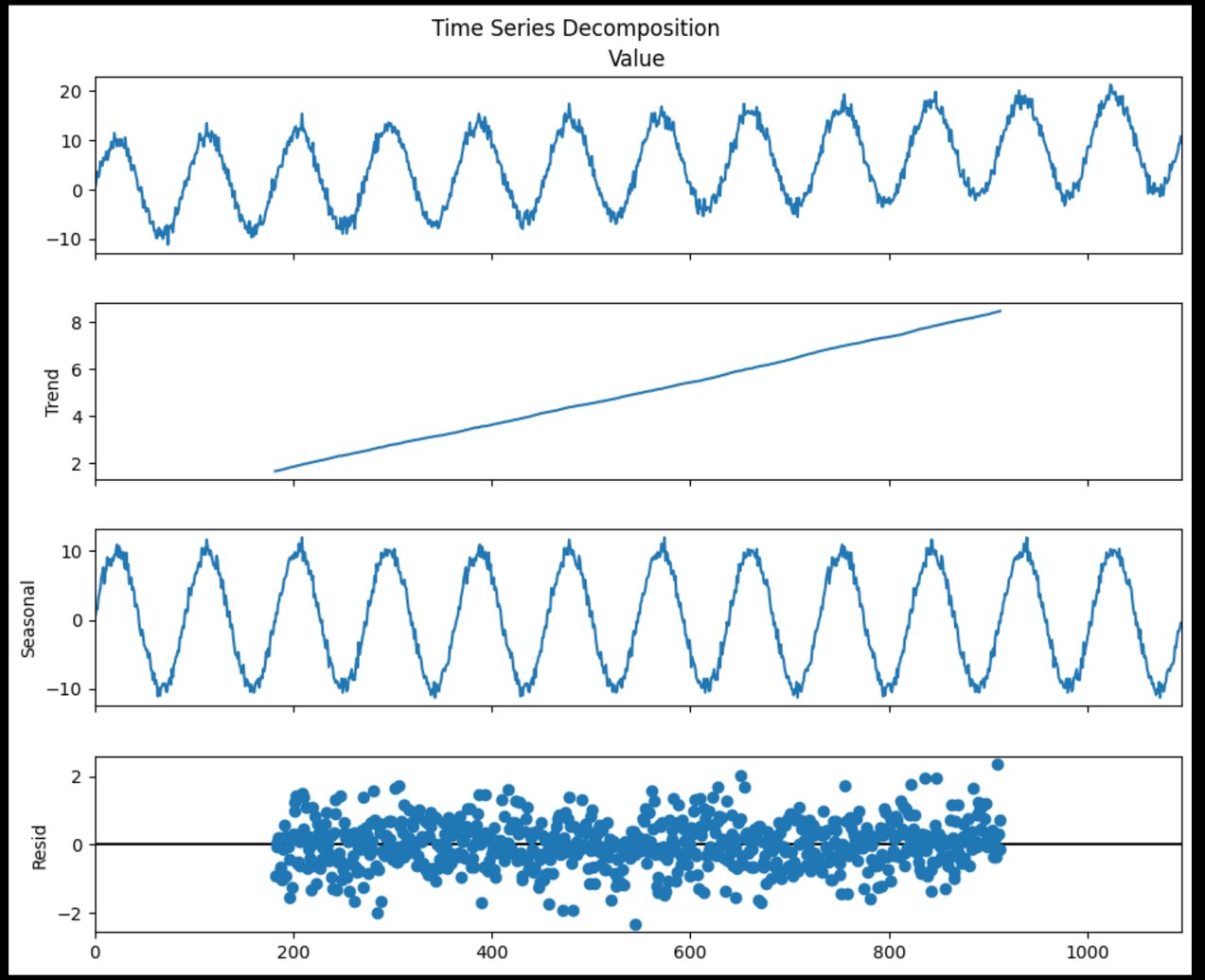
- Linear: $f(X) = WX + b$
- Sine: $f(t) = A \cdot \sin(\omega t + \phi)$ with A for amplitude, ω for angular frequency (radians/sec), and ϕ for phase shift with $0 \leq \phi \leq 2\pi$
- Exponential: $f(x) = e^x$
- Logistic: $f(x) = \dfrac{L}{1 + e^{-k(x-x_0)}}$ with L max value, k growth rate and x_0 midpoint



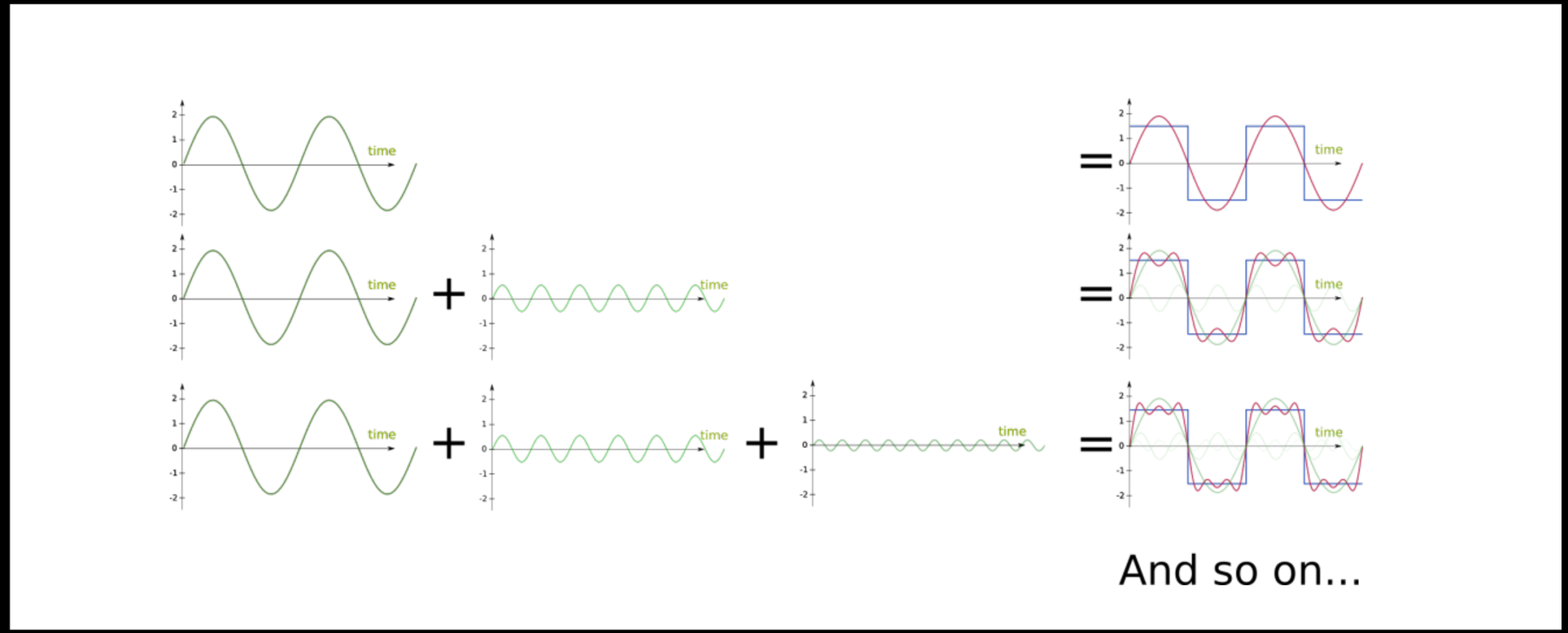




Timeseries decomposition



Fourier Transforms



Assumptions

What are outliers?

Assumptions

- Assume your data follows a normal distribution, are at least is approximately symmetric and not skewed.
- Assume your data is continous numerical data. Categorical, count or discretised data might not work well.
- Assume the data comes from a single population. Multimodal distributions arent handled well (what appears an outlier might be a different subpopulation).
- Assume observations are indepent of each other. It doesnt account for data with temporal, spatial, or hierarchical dependencies.

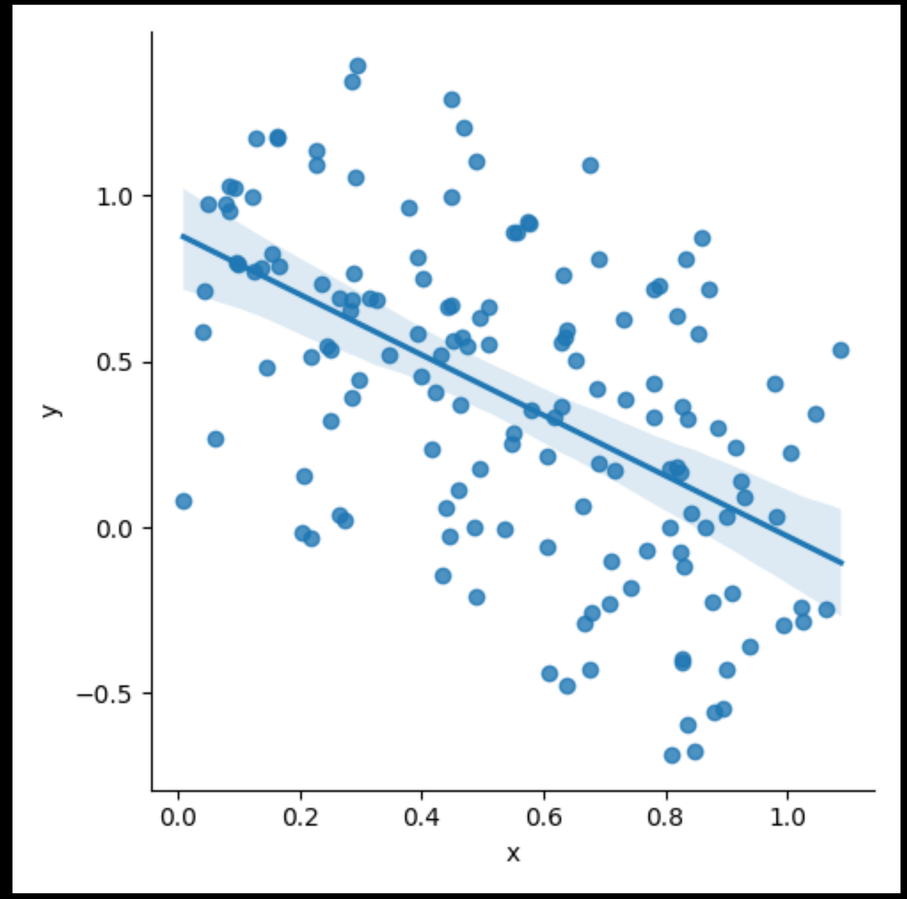
How does this translate to modelling?

Assumptions

- Dont assume your data follows a normal distribution; check if it is symmetric, skewed, and do statistical tests.
- Dont assume your data is continuous; it might be better modeled with a discrete distribution.
- Dont assume your data comes from a single population. Think about possible subpopulations.
- Check if your observations are indepent of each other. Think about temporal, spatial, or hierarchical dependencies.

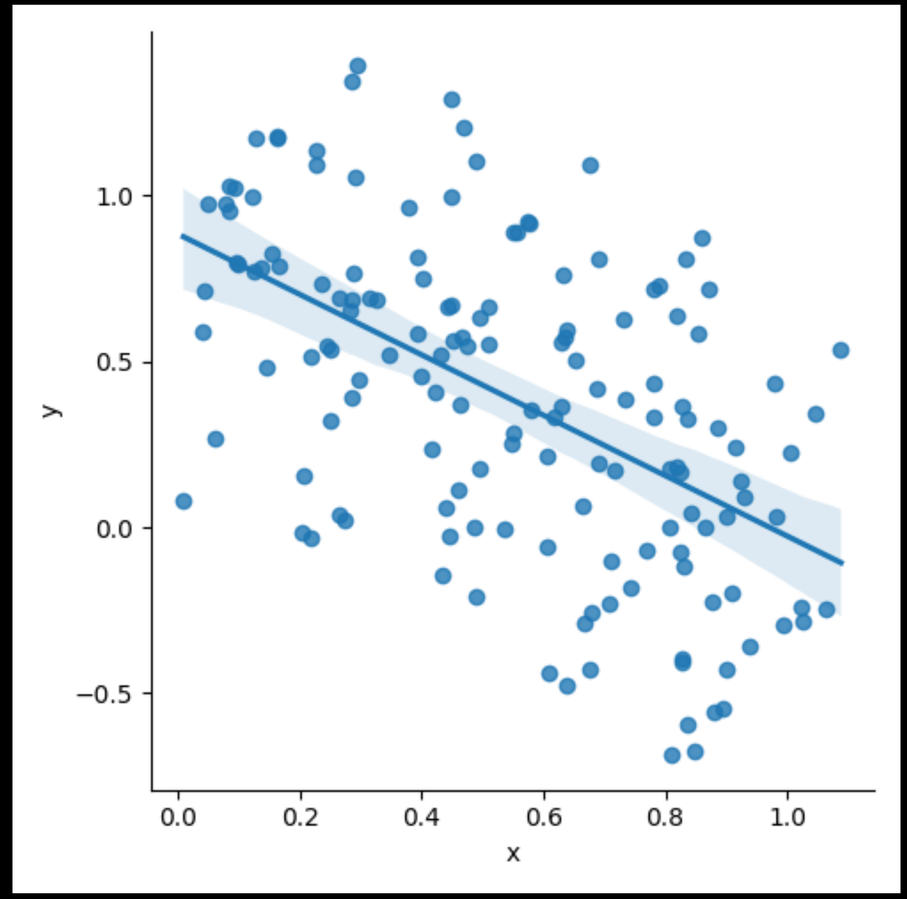
The Simpsons paradox

- The shaded area is the 99% confidence interval of the linear regression.
- What is your conclusion about the data?



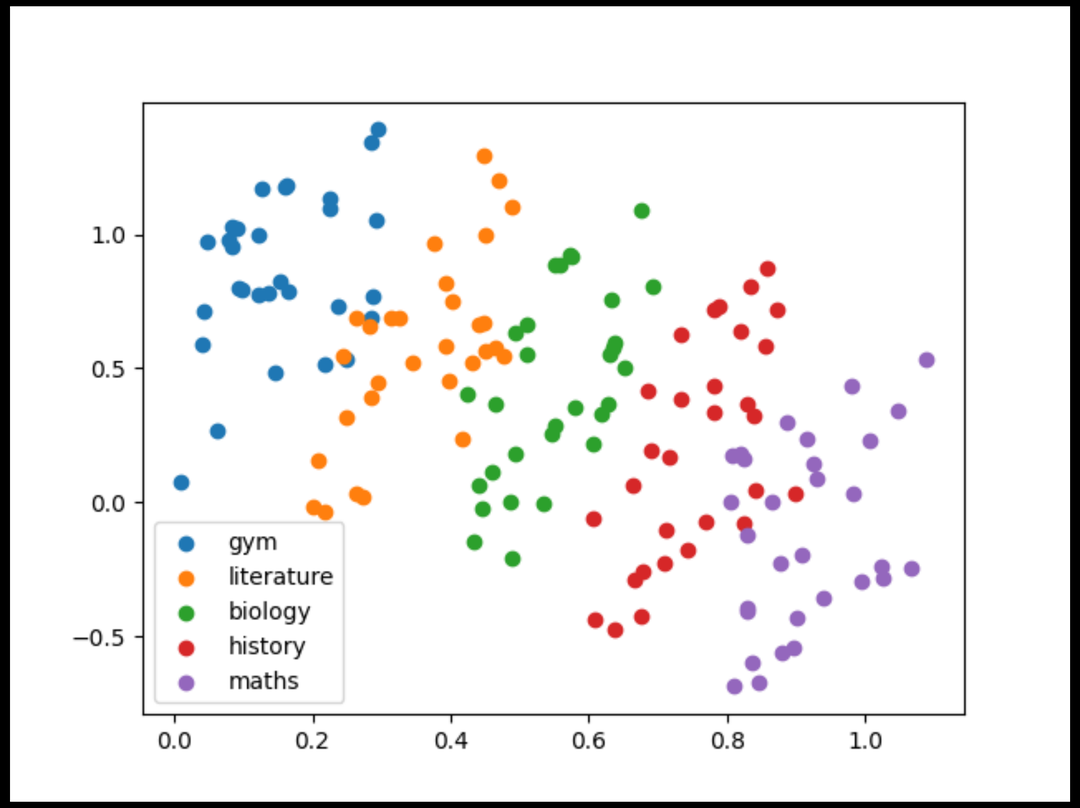
The Simpsons paradox

- The shaded area is the 99% confidence interval of the linear regression.
- Does your conclusion change if I tell you that the x axis is the amount of hours invested in study, and the y axis is the average grade of a student? Why?



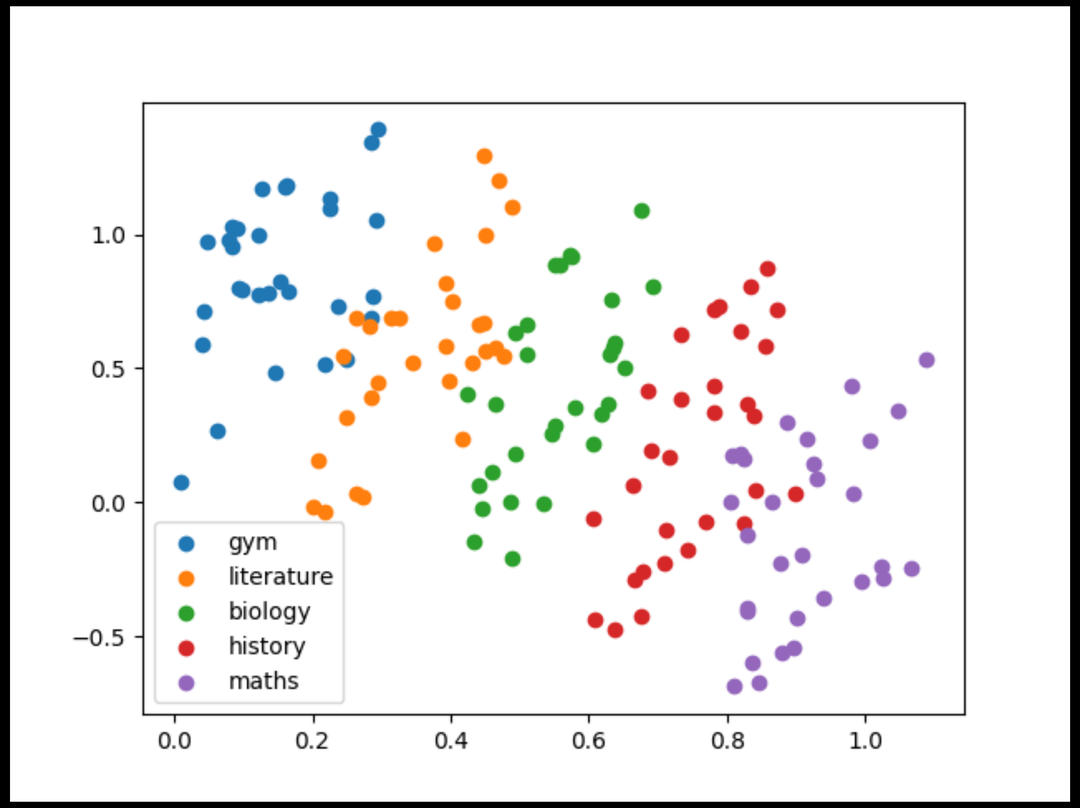
The Simpsons paradox

- The shaded area is the 99% confidence interval of the linear regression.
- the x axis is the amount of hours invested in study, and the y axis is the average grade of a student
- Does changing the colors change your conclusion?
- If so, should you have changed your inital conclusion, even without this extra information?



The Simpsons paradox

- Simpson's paradox is a phenomenon in which a trend appears in several groups of data but **disappears** or **reverses** with different groups.
- This result is often encountered in social-science and medical-science statistics
- It is particularly problematic when frequency data are undeservedly given **causal interpretations**



The Simpsons paradox

UC Berkely Gender Bias (admission fall 1973)

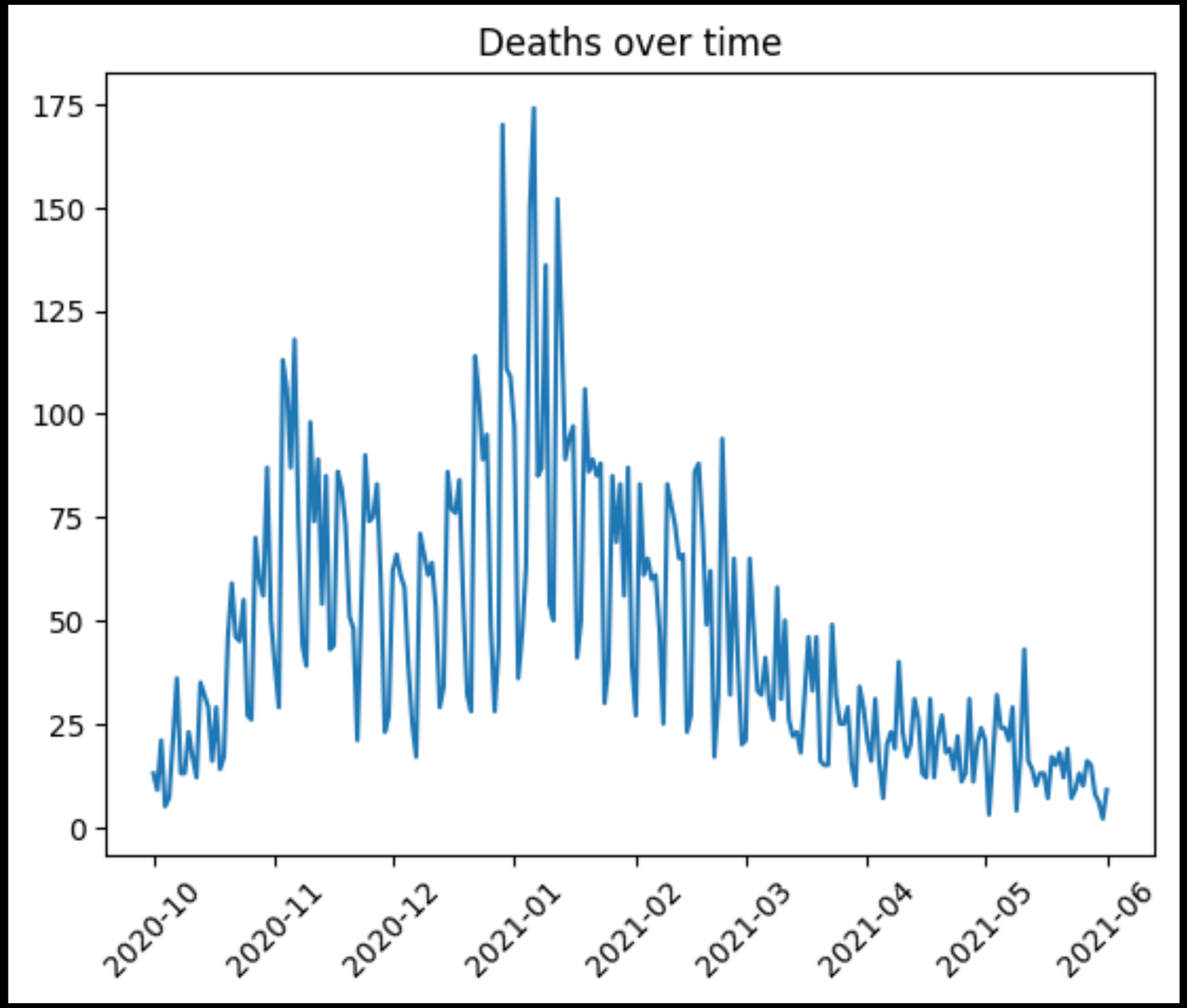
	All		Men		Women	
	Applicants	Admitted	Applicants	Admitted	Applicants	Admitted
Total	12,763	41%	8,442	44%	4,321	35%

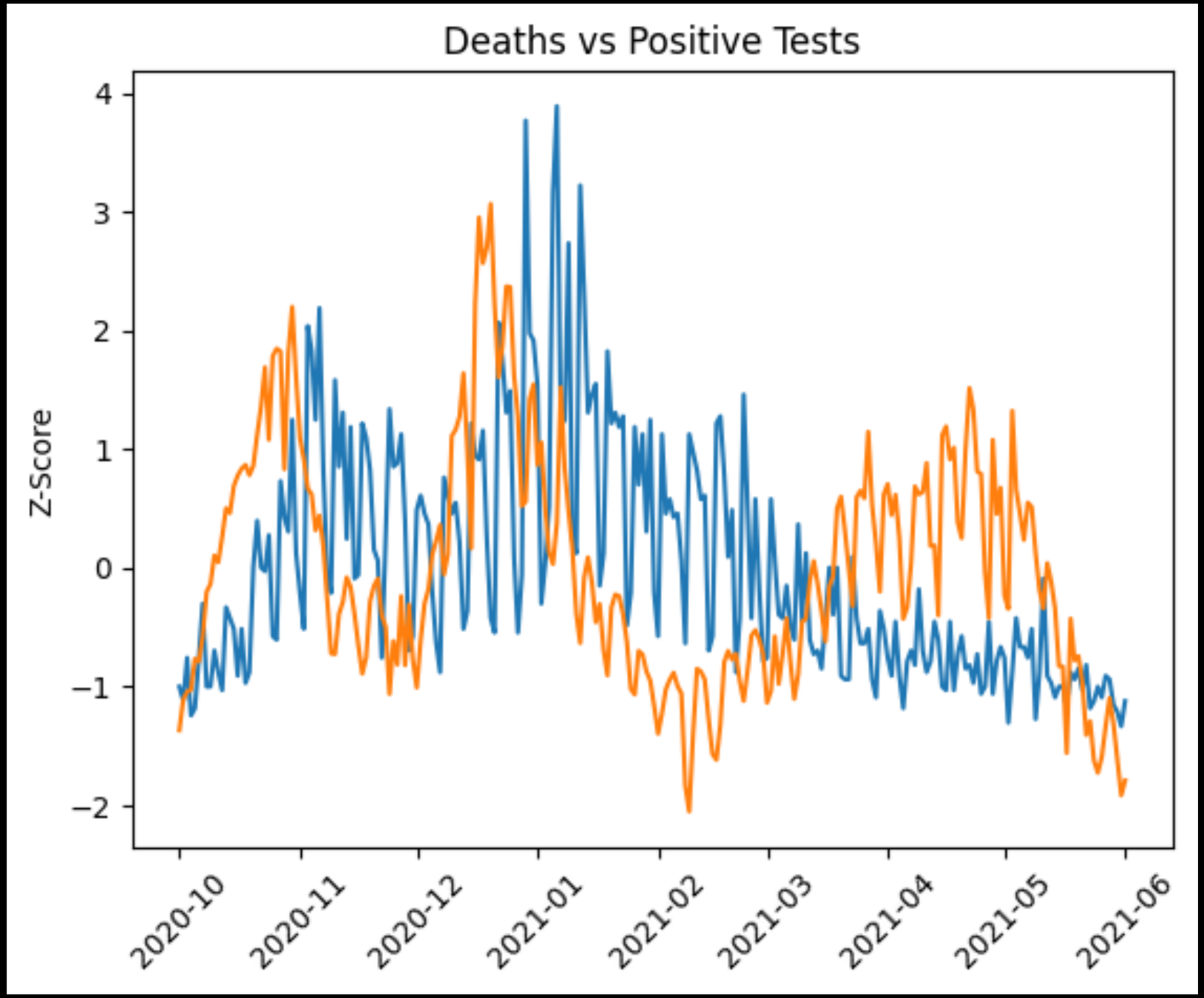
The Simpsons paradox

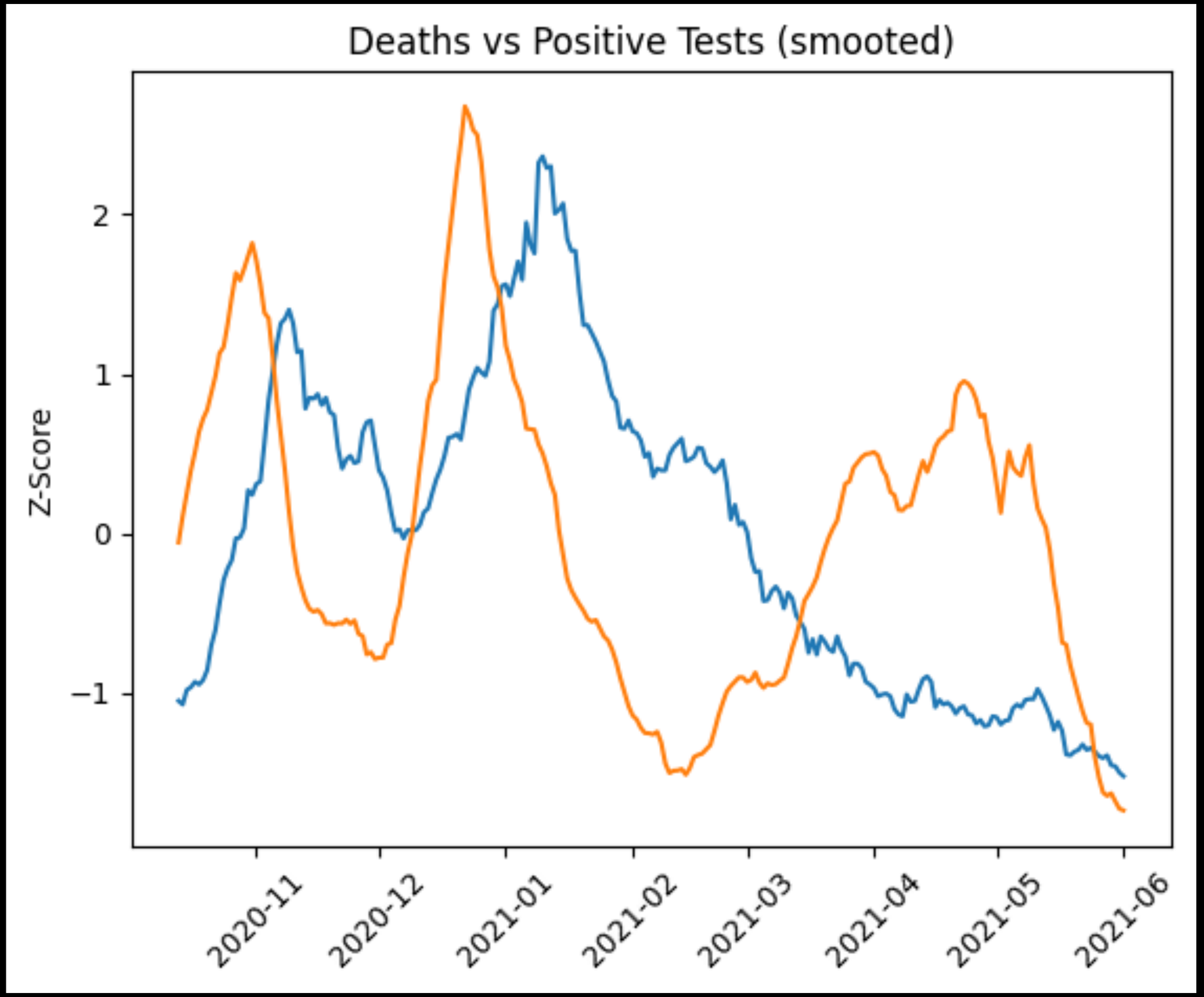
UC Berkely Gender Bias (admission fall 1973)

Department	All		Men		Women	
	Applicants	Admitted	Applicants	Admitted	Applicants	Admitted
A	933	64%	**825**	62%	108	82%
B	585	63%	**560**	63%	25	68%
C	918	35%	325	37%	**593**	34%
D	792	34%	417	33%	375	35%
E	584	25%	191	28%	**393**	24%
F	714	6%	373	6%	341	7%
Total	**4526**	**39%**	**2691**	**45%**	**1835**	**30%**

Case: modelling corona


Deaths over time
175
150
125
100
75
50
25
0
2020-10
2020-11
2020-12
2021-01
2021-02
2021-03
2021-04
2021-05
2021-06


Deaths vs Positive Tests
Z-Score
4
3
2
1
0
−1
−2
2020-10
2020-11
2020-12
2021-01
2021-02
2021-03
2021-04
2021-05
2021-06


Deaths vs Positive Tests (smooted)
Z-Score
2
1
0
−1
2020-11
2020-12
2021-01
2021-02
2021-03
2021-04
2021-05
2021-06

Coronavaccin

NOS Nieuws • Dinsdag 5 januari 2021, 11:53

Coronavaccinatie begint: wanneer ben jij aan de beurt (als de planning uitkomt)?

De eerste mensen in Nederland worden vanaf morgen ingeënt tegen het coronavirus. In Veghel krijgen verpleeghuismedewerkers hun eerste prik met het Pfizer/BioNTech-vaccin, vooralsnog het enige vaccin dat is goedgekeurd in de EU. Wanneer volgt de rest?

Our primary data

- Positive tests $X = \{x_1, \ldots, x_n \mid x_i \in \mathbb{R}\}$
- Deaths $y = \{y_1, \ldots, y_n \mid y_i \in \mathbb{R}\}$

$$y_t = ax_{t-14}$$

- What does b represent?

$$y_t = ax_{t-14} + b$$

- And ϵ ?
- Why is that different from b?

$$y_t = ax_{t-14} + b + \epsilon$$

$$y_t = ax_{t-14} + b + \epsilon$$

$$\epsilon \sim \mathcal{N}(0,\sigma)$$

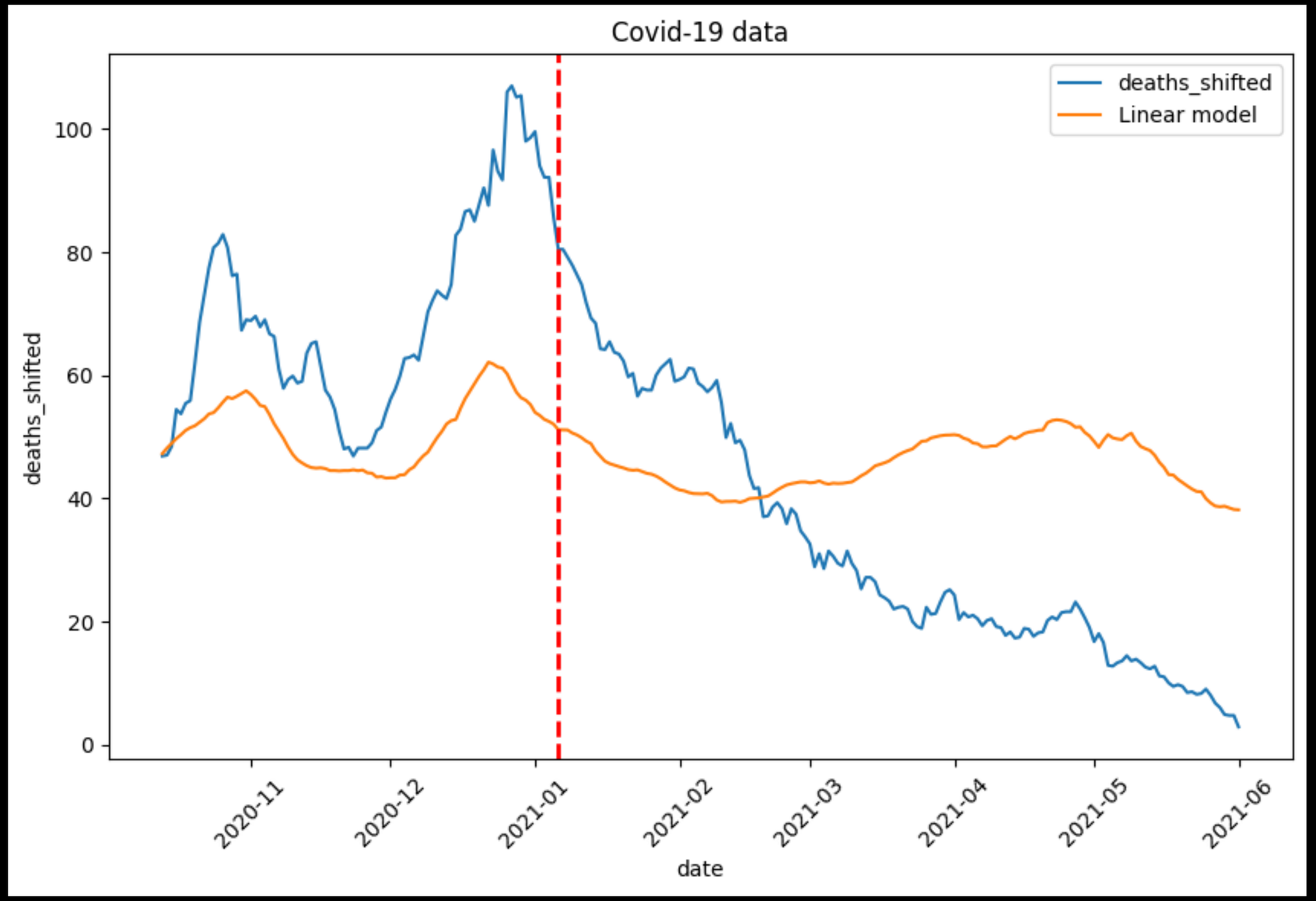
- How can you estimate b? And ϵ?

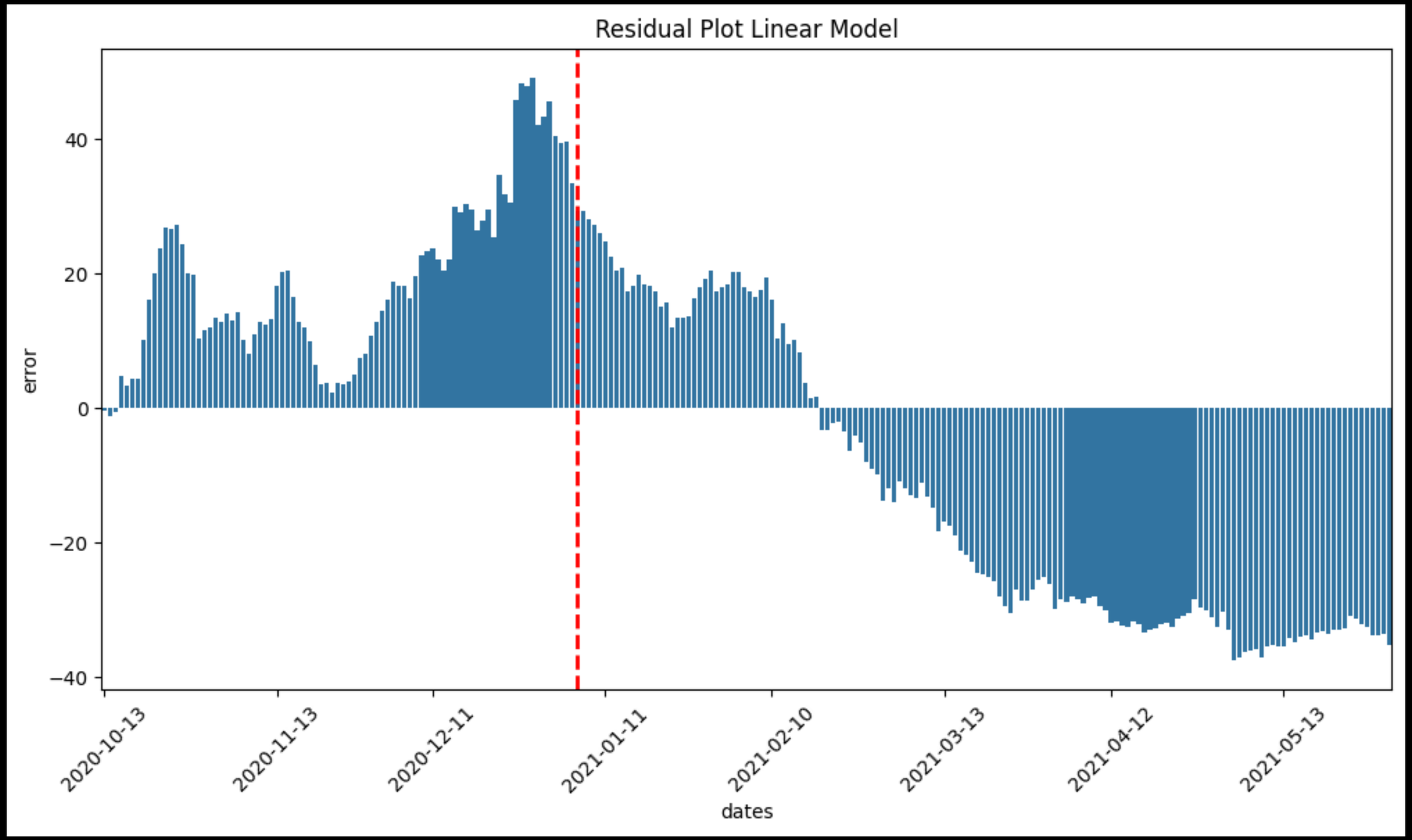
$$y_t = ax_{t-14} + b + \epsilon$$

$$\epsilon \sim \mathcal{N}(0,\sigma)$$

$$y_t - (ax_{t-14} + b) = \epsilon$$

Linear model results



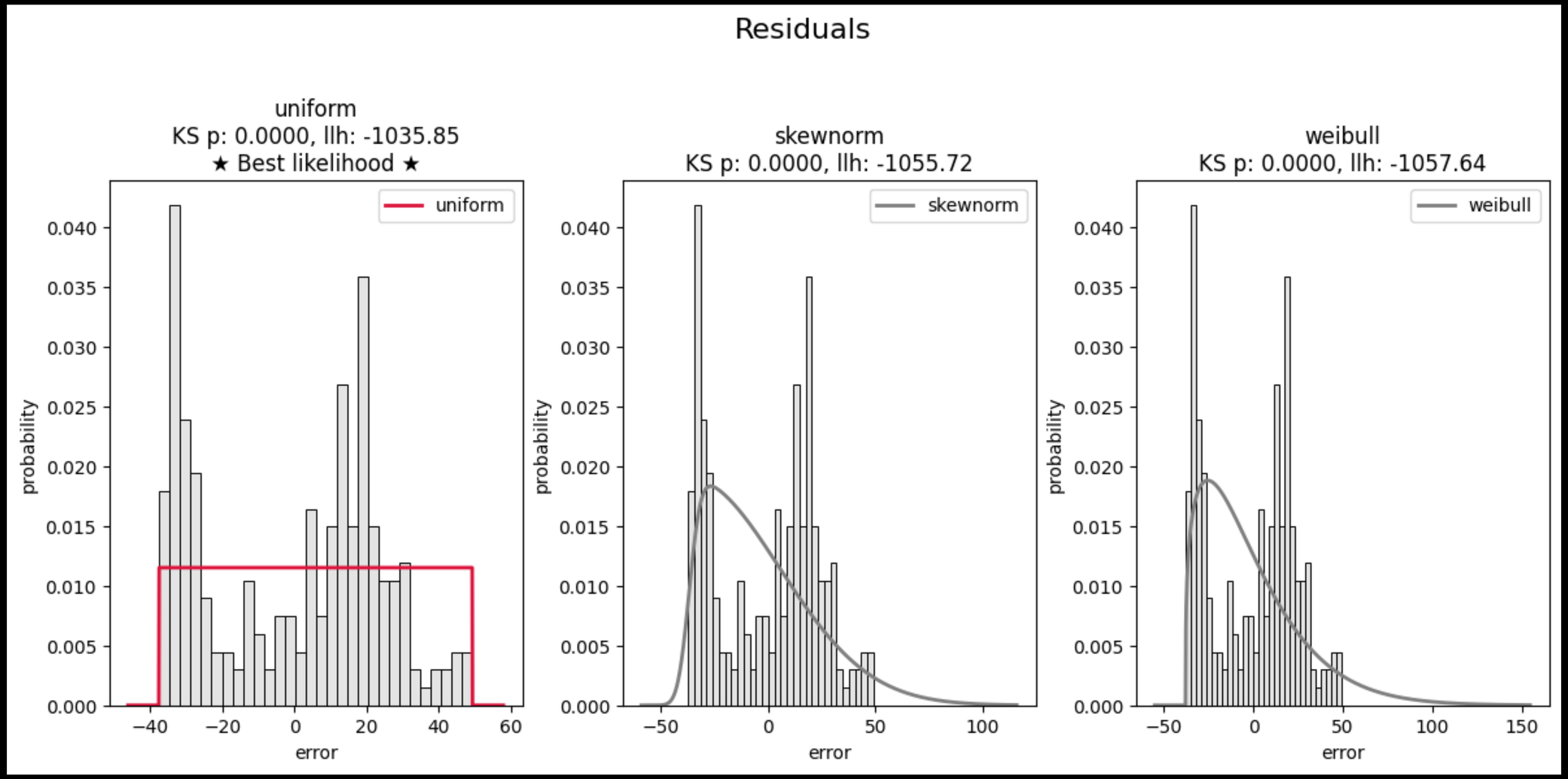

Residual Plot Linear Model
error
40
20
0
−20
−40
2020-10-13
2020-11-13
2020-12-11
2021-01-11
2021-02-10
2021-03-13
2021-04-12
2021-05-13
dates

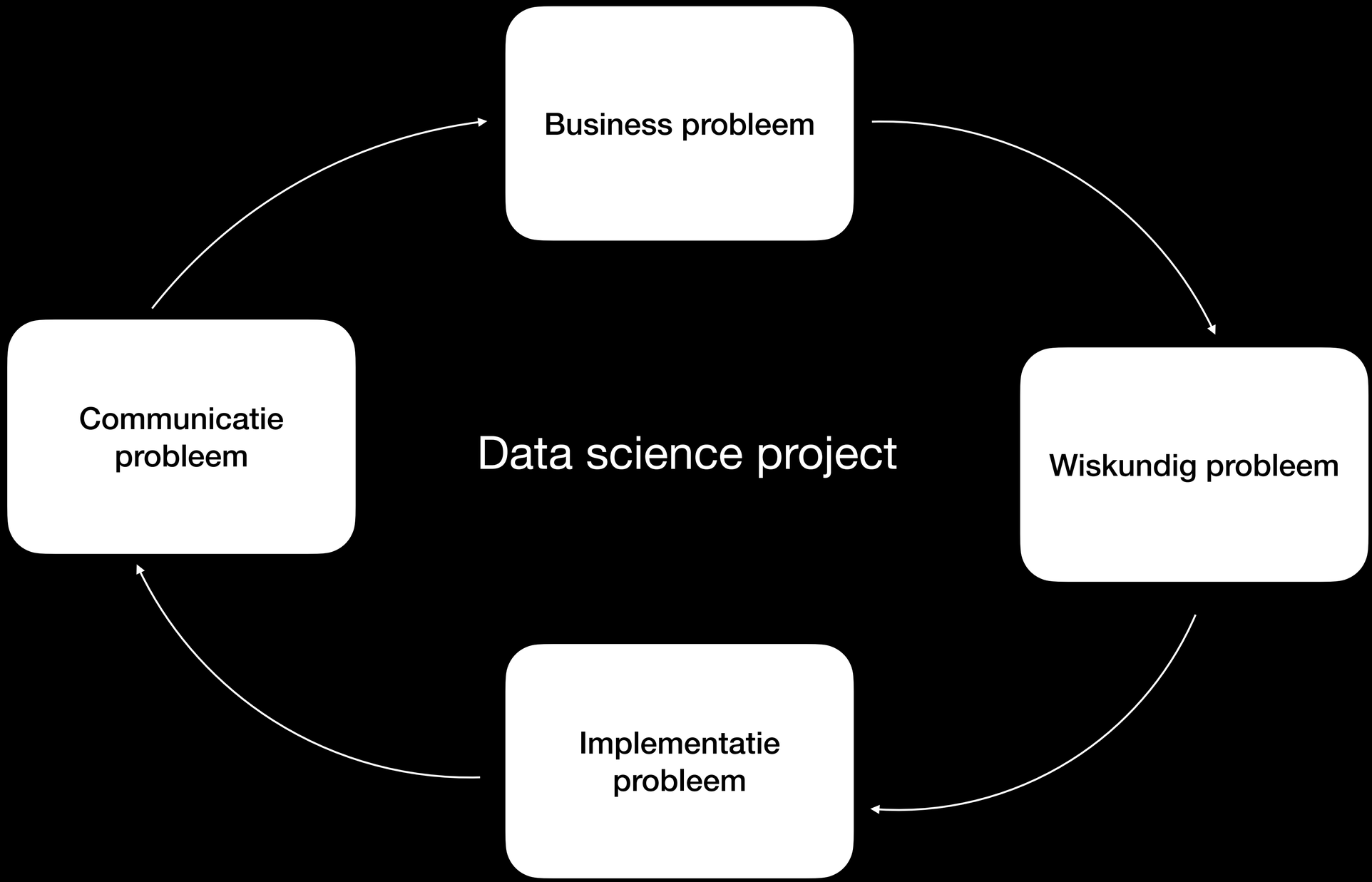
About p-values

- The p-value represents the probability of observing your data (or more extreme data) if the null hypothesis were true.
- When $p<0.05$, it means there's less than a 5% chance that the pattern you observed would occur if the null hypothesis were actually true.
- This 5% threshold is a conventionally accepted level of risk for a Type I error (incorrectly rejecting a true null hypothesis).
- When the p-value falls below this threshold, statisticians traditionally consider the evidence strong enough to reject the null hypothesis in favor of the alternative hypothesis.

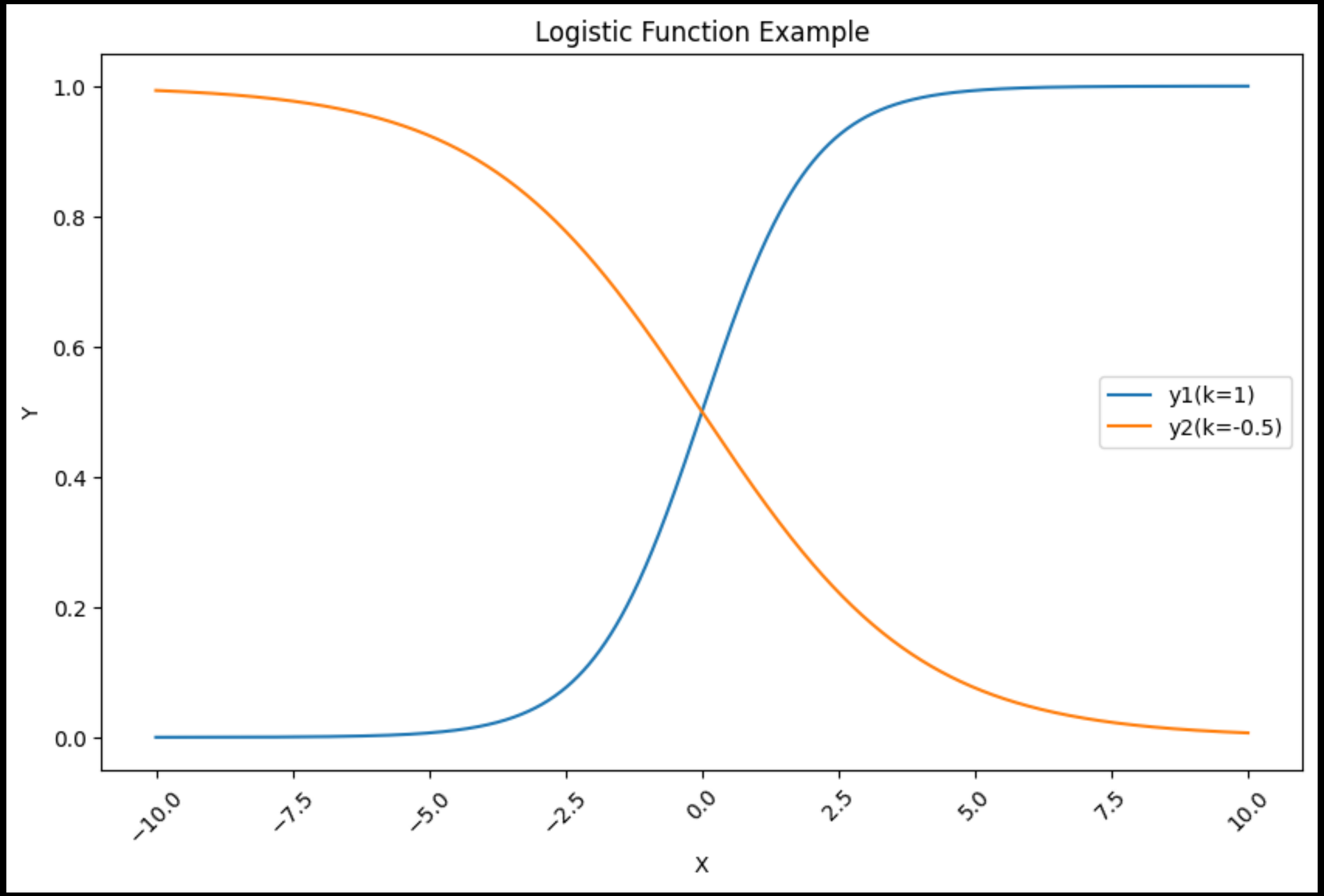
Testing the residual

Null-hypothesis is distributions are equal



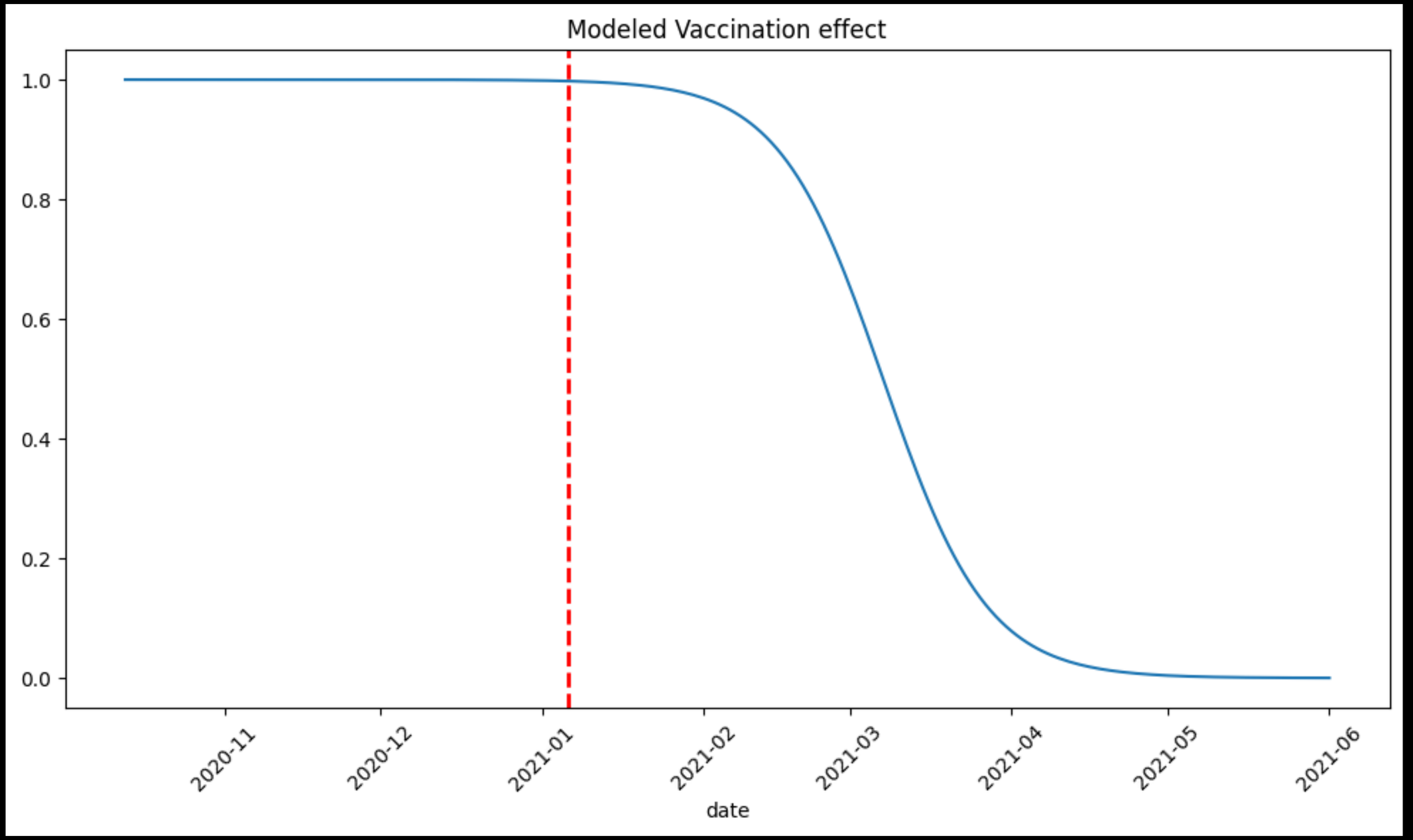

Business probleem
Wiskundig probleem
Implementatie probleem
Communicatie probleem
Data science project

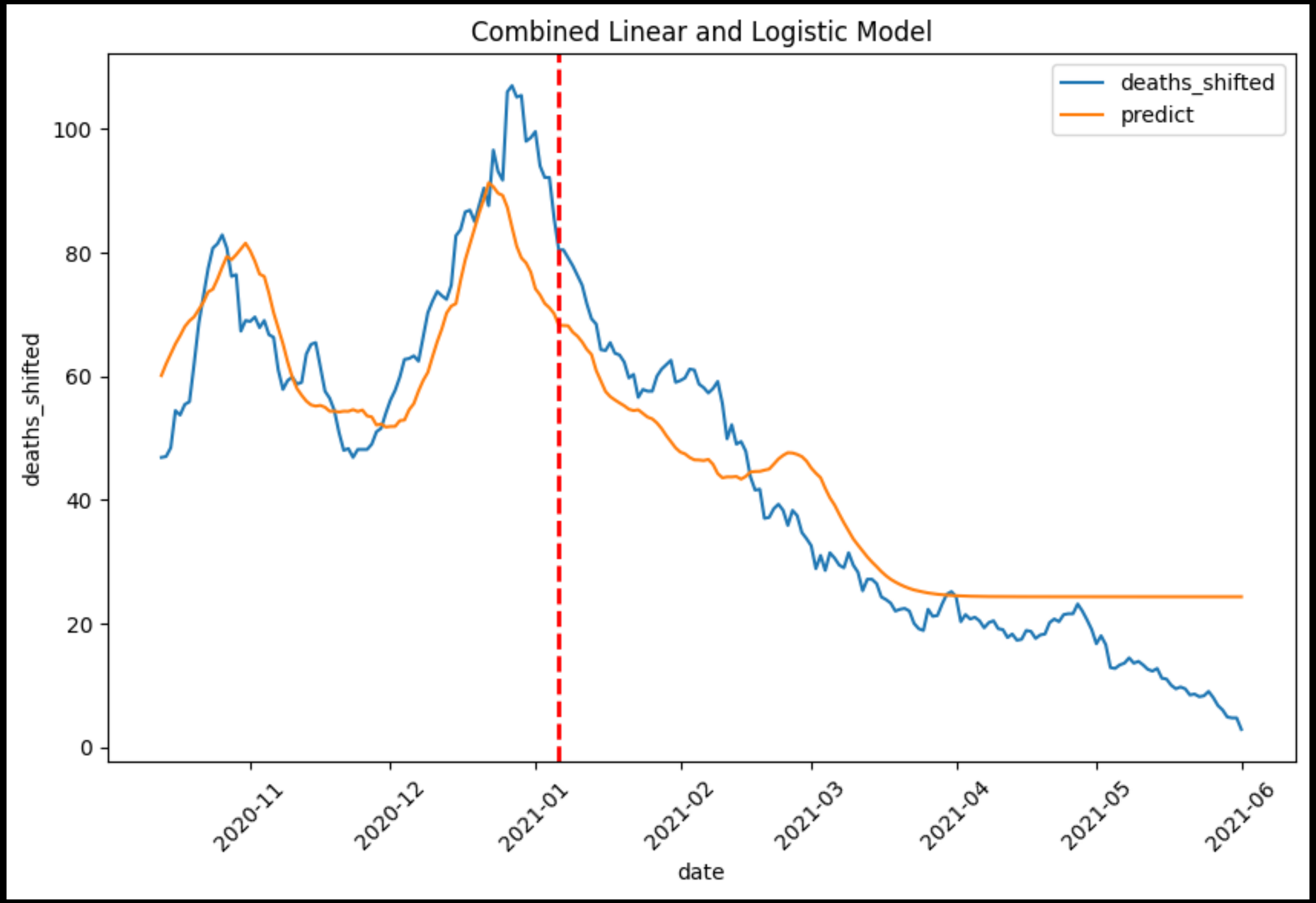
$$f(x) = \frac{L}{1 + e^{-k(x-x_0)}}$$


Logistic Function Example
1.0
0.8
0.6
0.4
0.2
0.0
Y
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
X
y1(k=1)
y2(k=-0.5)

$$v = logistic_{\theta}(t)$$

$$y_t = vax_{t-14} + b + \epsilon$$


Modeled Vaccination effect
1.0
0.8
0.6
0.4
0.2
0.0
2020-11
2020-12
2021-01
2021-02
2021-03
2021-04
2021-05
2021-06
date


Combined Linear and Logistic Model
deaths_shifted
predict
100
80
60
40
20
0
deaths_shifted
2020-11
2020-12
2021-01
2021-02
2021-03
2021-04
2021-05
2021-06
date

https://github.com/raoulg/goad_exercises