

---

# Trainieren Neuronaler Netze zur Klimavorhersage

- Torge Schwark, Joschua Quotschalla

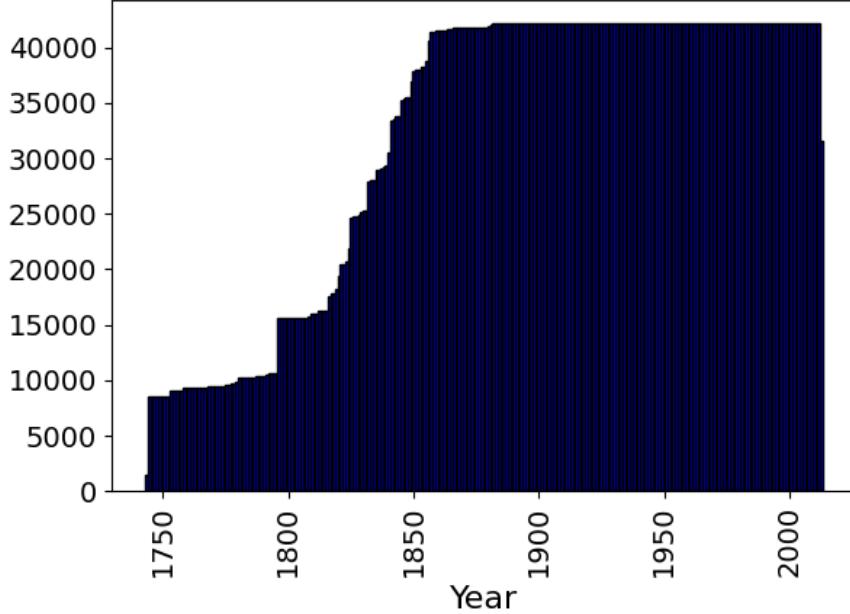
# Ablauf

---

- Datensatz: Climate Change
- Vorgehensweise (Datenaufarbeitung, Data-pipeline)
- Was ist eine gute Vorhersage?
- Grid Search
- Fine Tuning
- Klima Vorhersagen

Data Amount by Year

Amount of data



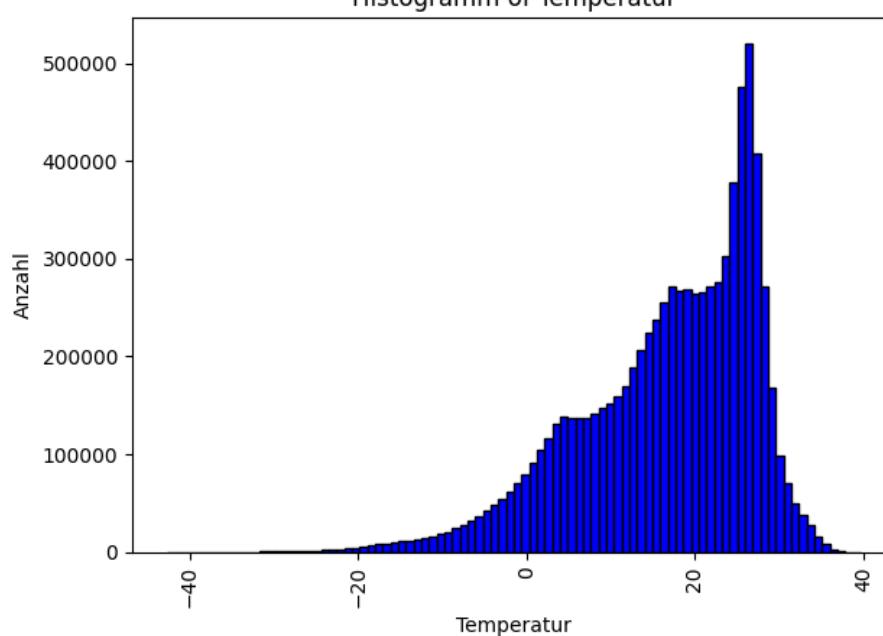
# Climate Change: Earth Surface Temperature Data

Monatliche Temperaturdaten von über 4000 Standpunkten

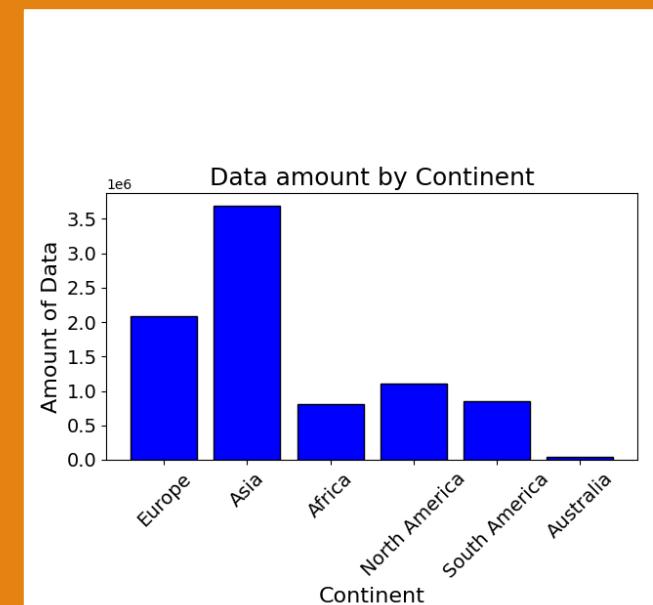
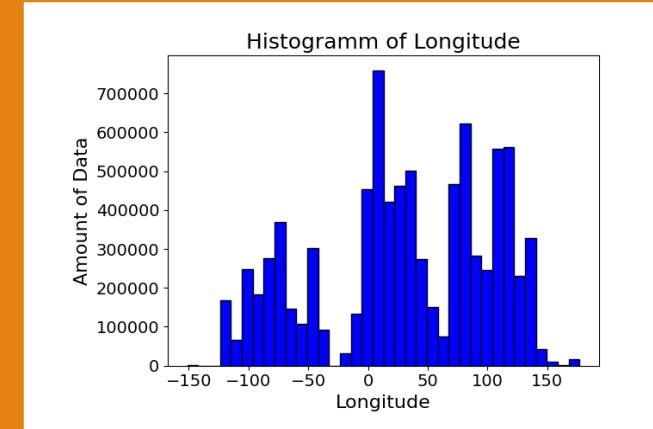
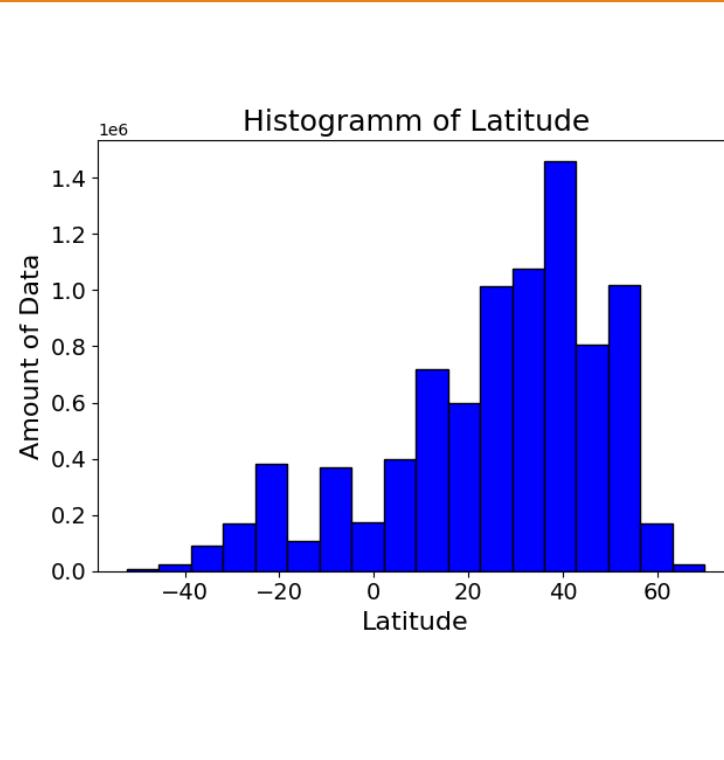
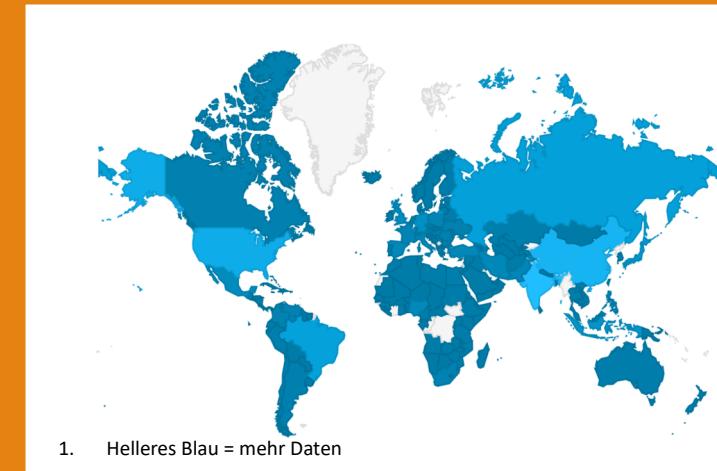
Seit bis zu 260 Jahren

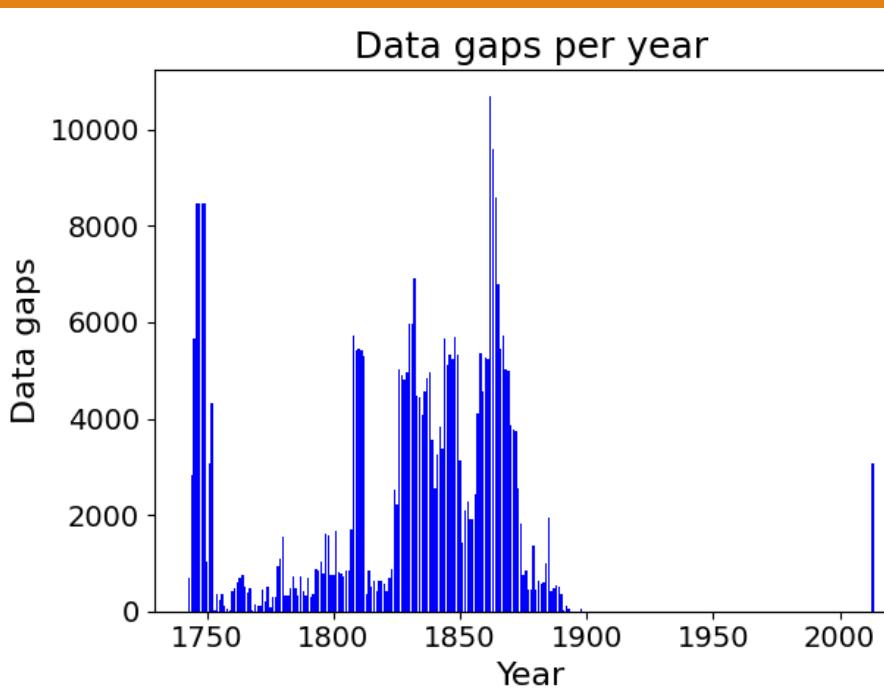
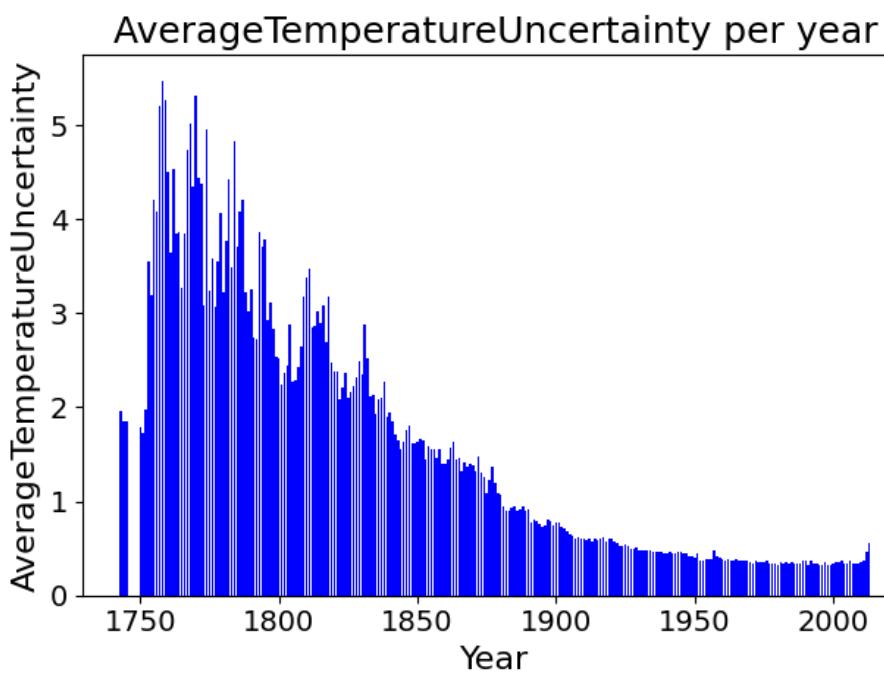
Mehr als 10.000.000 Datenpunkte

Histogramm of Temperatur



# Datensatz: Climate Change





# Datensatz: Climate Change

---

Daten enthalten Lücken und Ungenauigkeiten

Datensatz bestehend aus Daten von 16 bereits existierenden Archiven

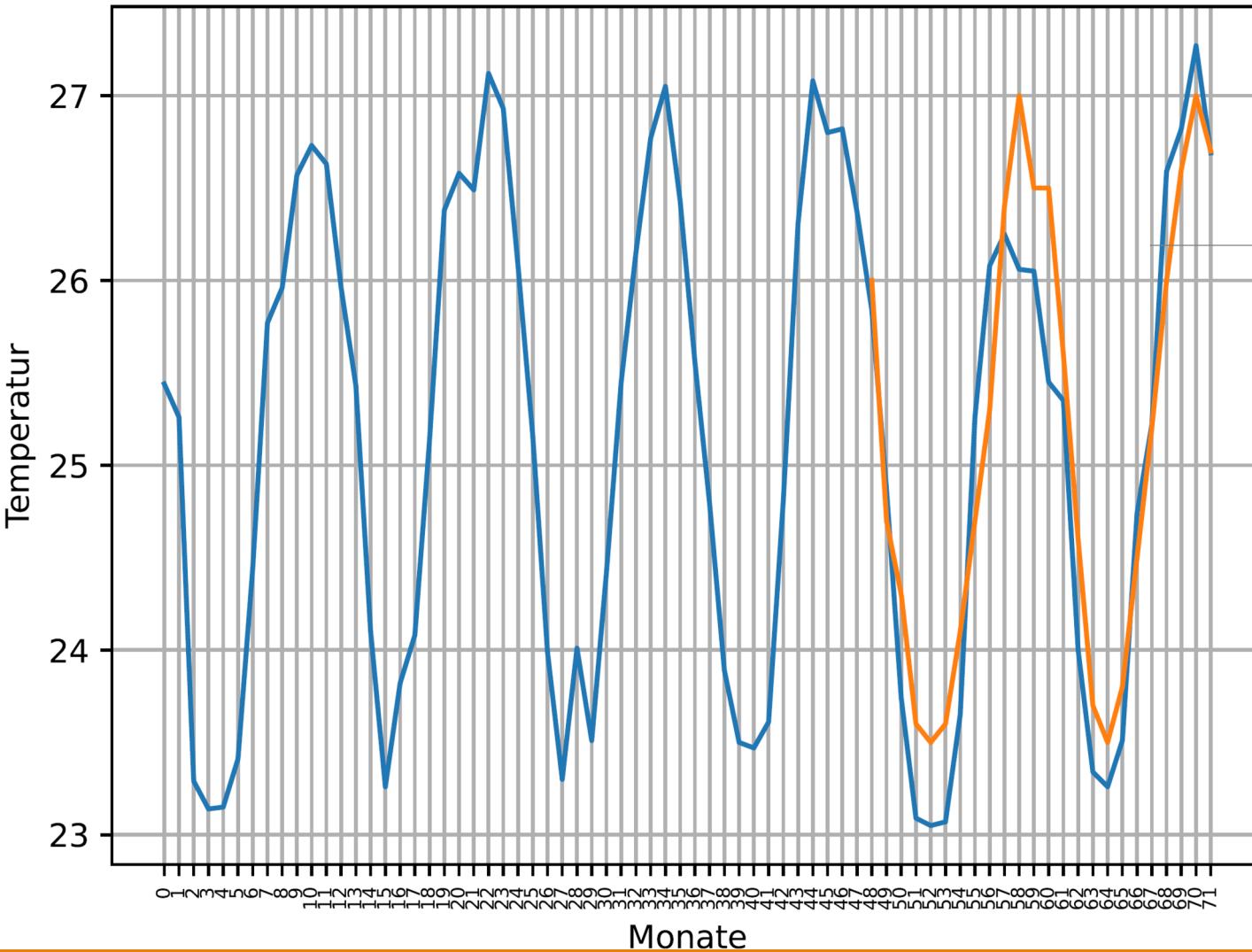
117 Duplikate (2 oder mehrfach)

# Datenaufbereitung, Data-pipeline

---

- Ausschließlich vollständige Daten
- 1. Versuch: Data-Loader angelehnt an Sinus Datensatz
- 2. Versuch: Daten in Listen Speichern (keine Verbesserung)
- 3. Multithreading mittels "Multiprocessing->Pool" (einfache aber große Verbesserung)

### Temperature



# Was ist eine gute Vorhersage?

Experiment: Wie gut können wir Klimavorhersagen treffen?

30 Standorte 2 Jahre Vorhersagen (720 Datenpunkte)

MSE: 7.86, MAE: 1.91°C

Ziel: Bessere Vorhersagen treffen

# Trainieren der Neuronalen Netze

---

- Input: 840 Monate (70 Jahre), Output: 300 Monate (25 Jahre)
- 30% der Daten zum Validieren
- 70% zum Trainieren
- Netzwerk-Architekturen: MLP, ConvNets , LSTM, Transformer

# Grid Search

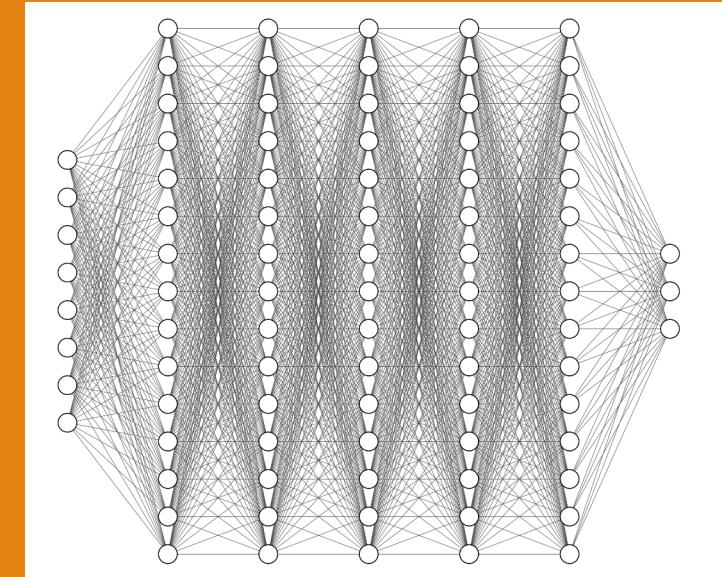
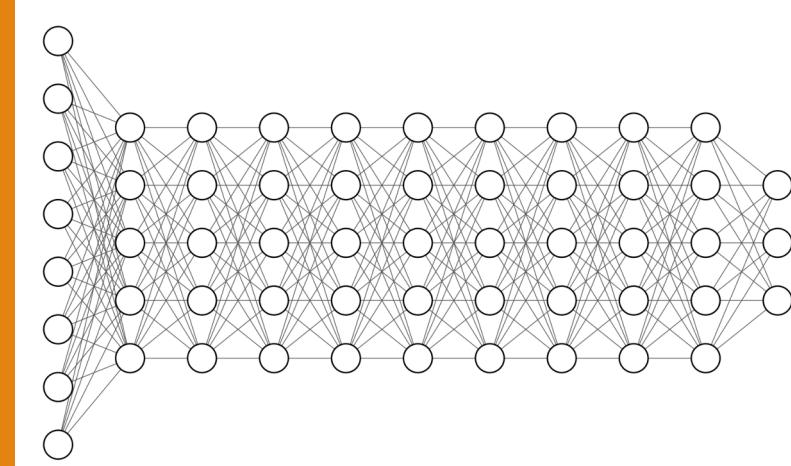
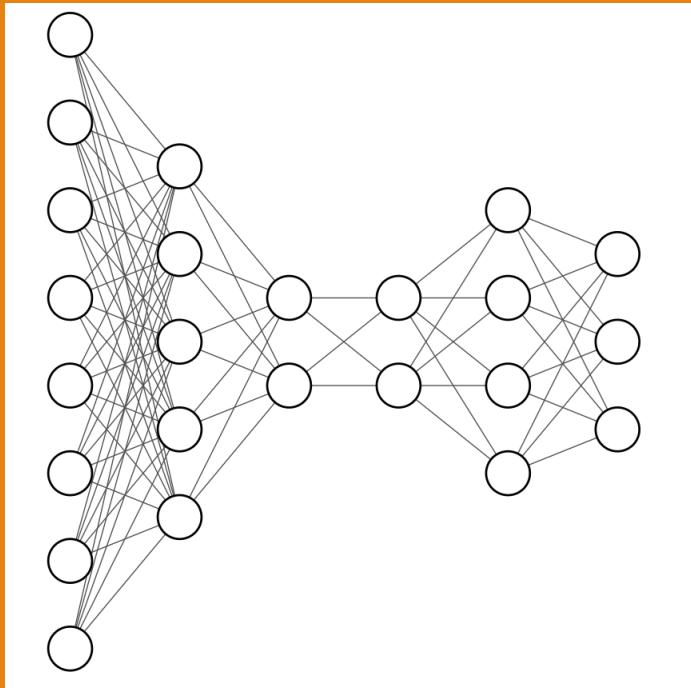
---

- 1.Schritt: Grid Search für MLP, ConvNets und LSTM
- 2.Schritt: Fine Tuning mit Data Augmentation, Dropout, Normalisieren, activation Function

## Grid Search:

- Unterschiedliche Architekturen, Learningrates, Batch sizes (Dropout)
- Fertig trainieren (Patience = 10)
- Anpassen der Steps per Epoch (vergleichbare Lerngeschwindigkeiten)
- MAE als Performance-Metrik

-

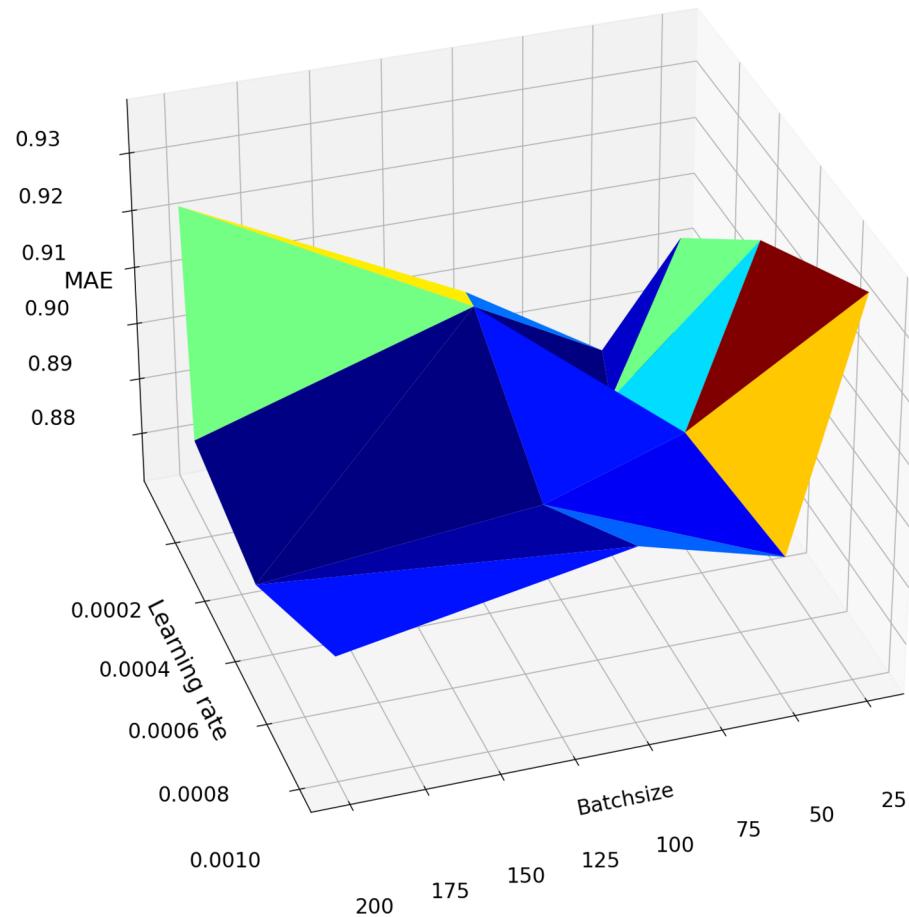


# Grid search MLP

# Grid search MLP

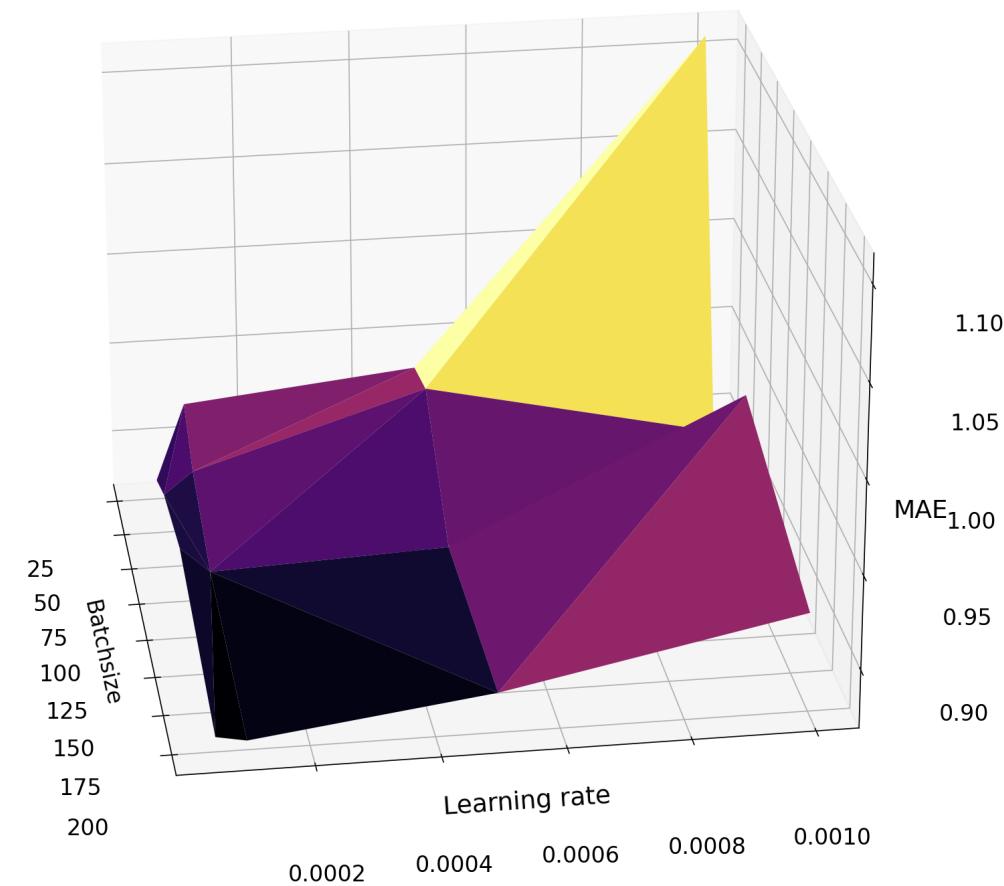
MAE arch compared

[500, 200, 200, 400]



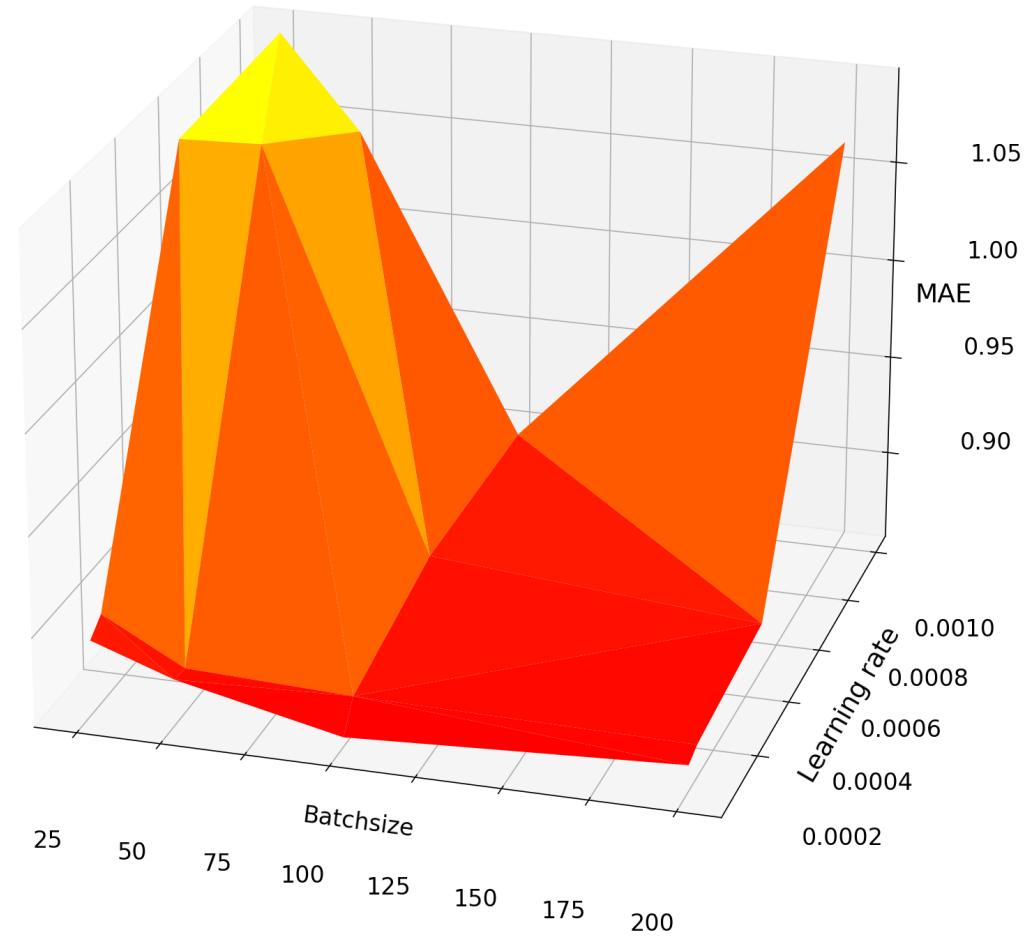
MAE arch compared

[1500, 1500, 1500, 1500, 1500]



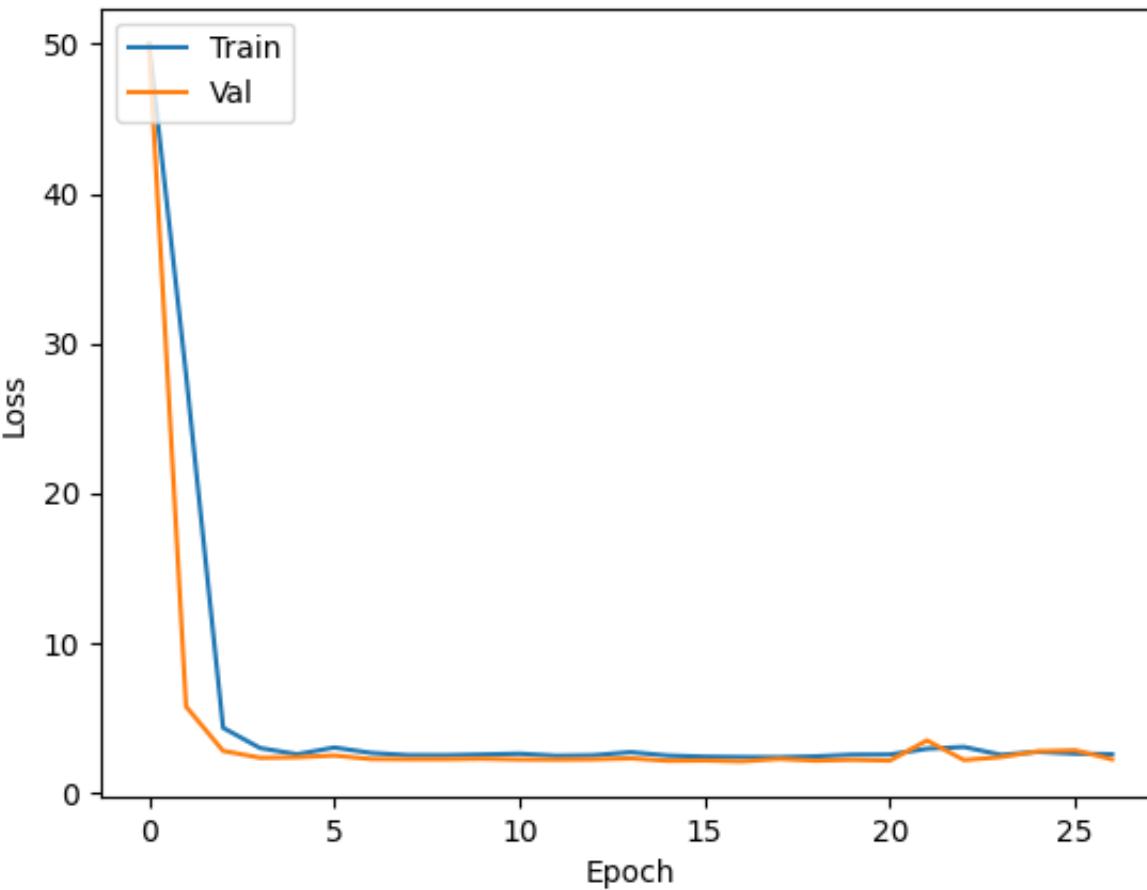
MAE arch compared

■ [500, 500, 500, 500, 500, 500, 500, 500, 500]



# Grid search MLP

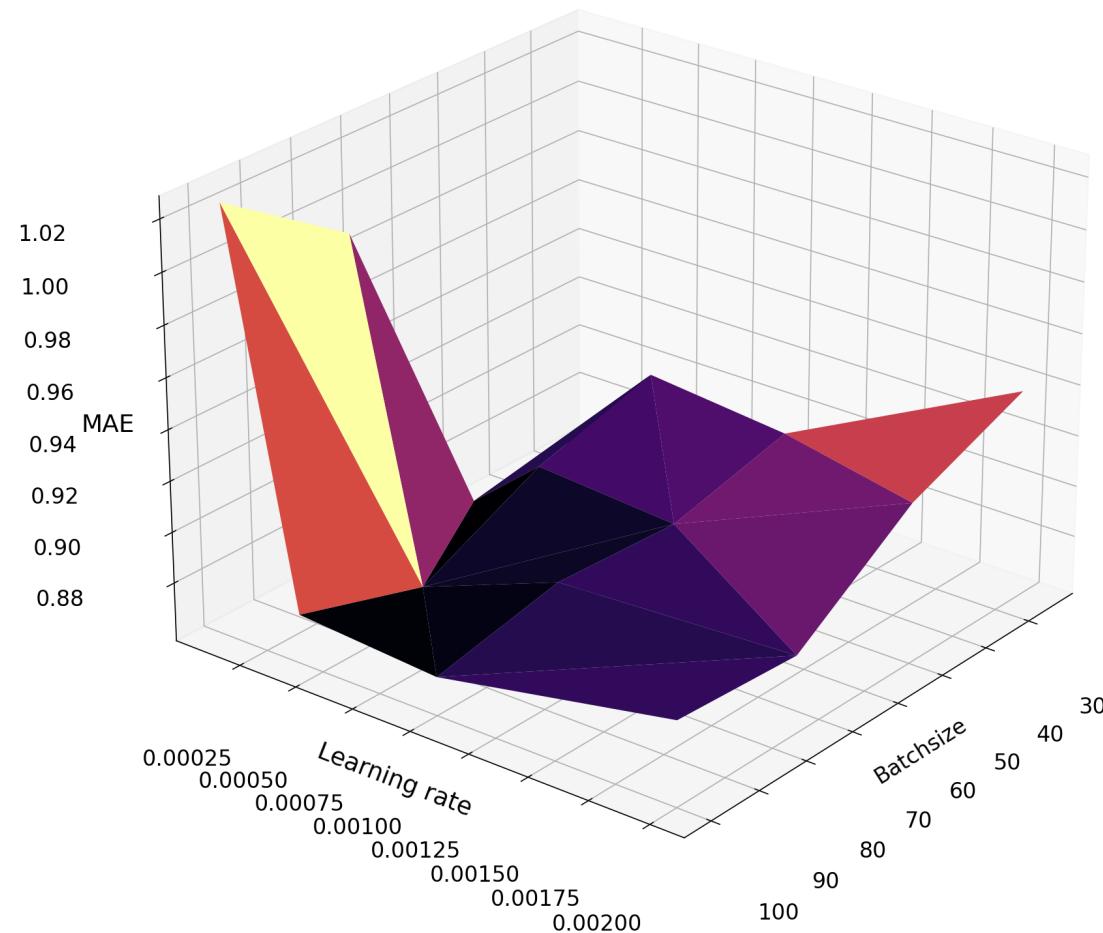
### Model Loss



## MLP Loss-Curve

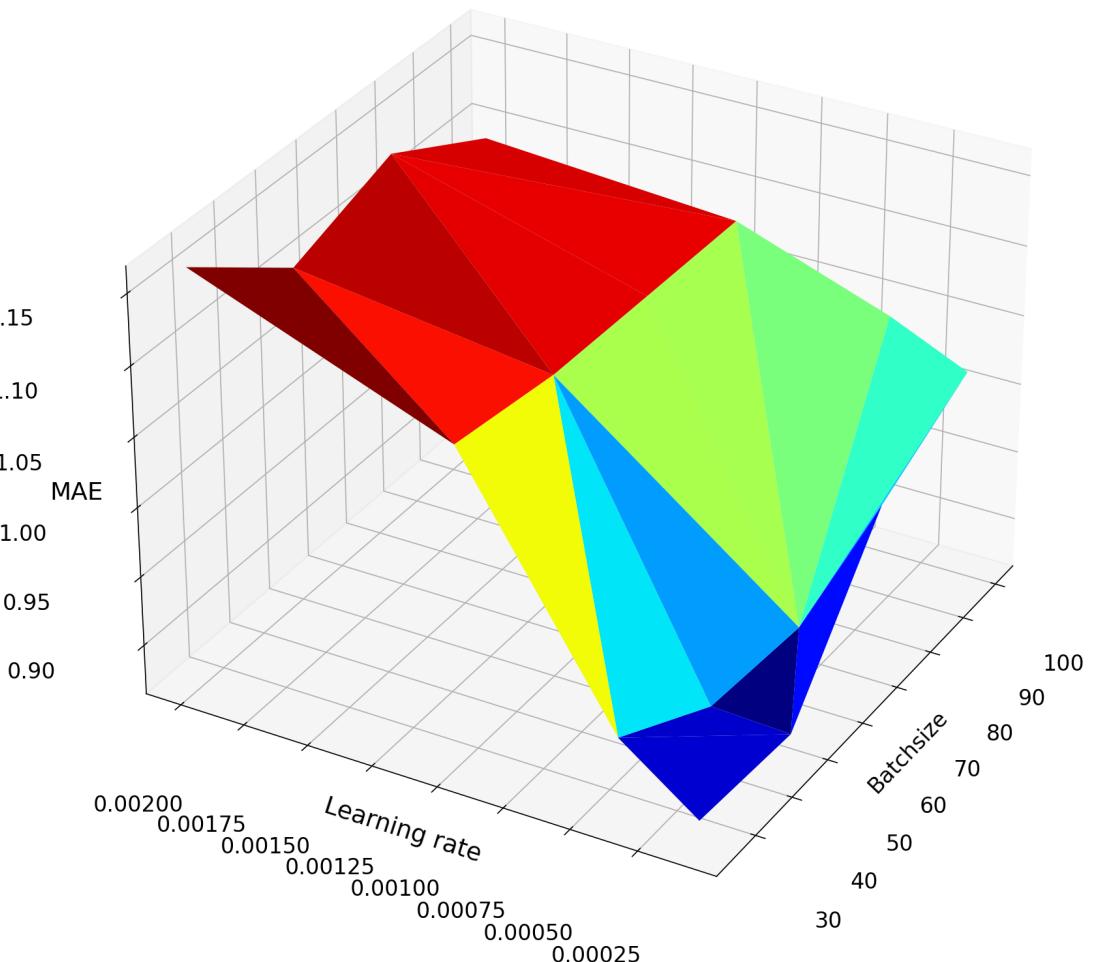
MAE arch compared

A legend box containing two entries: a blue square followed by the text "[[10, 10, 10], [5, 5, 5]]".



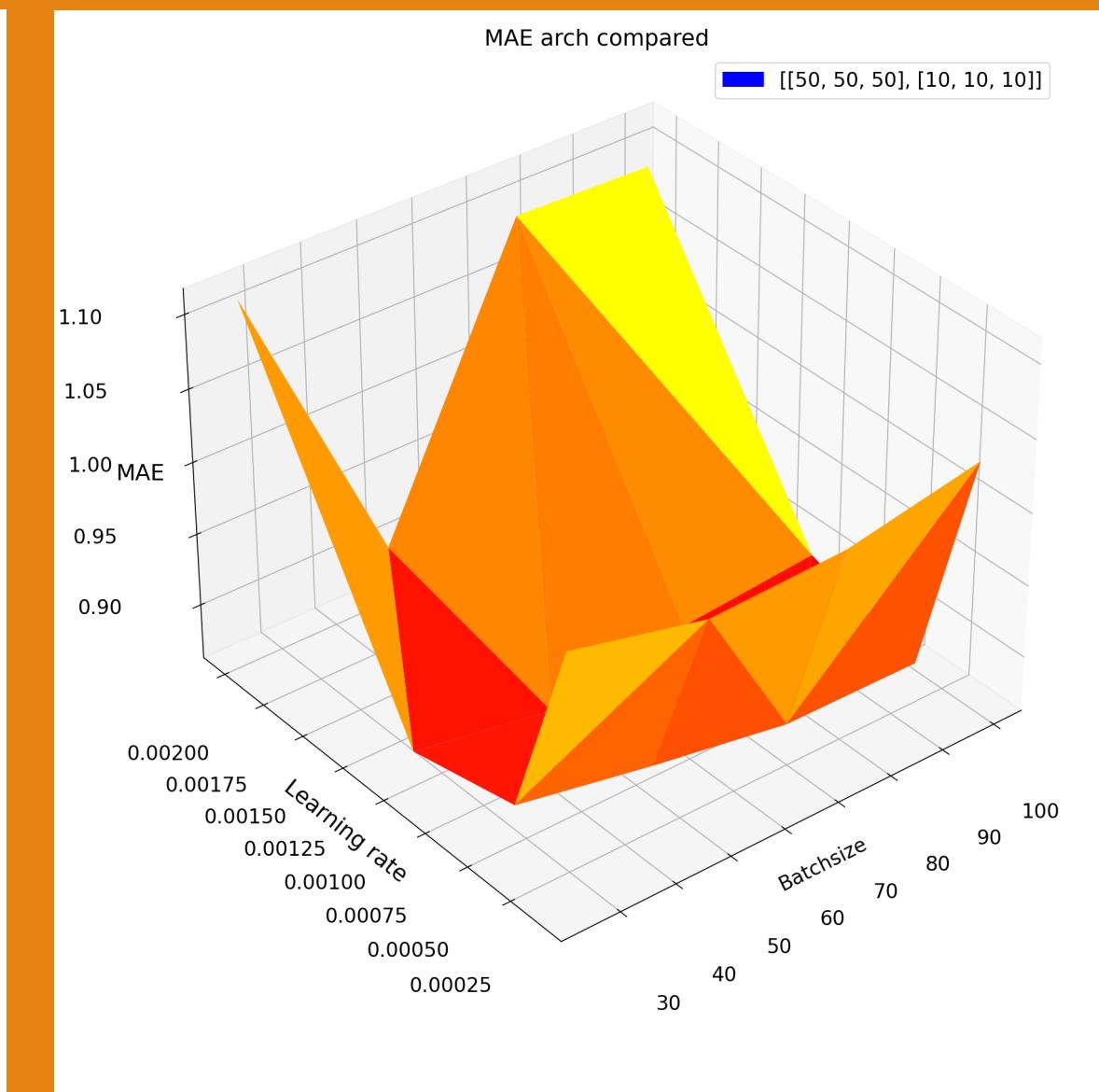
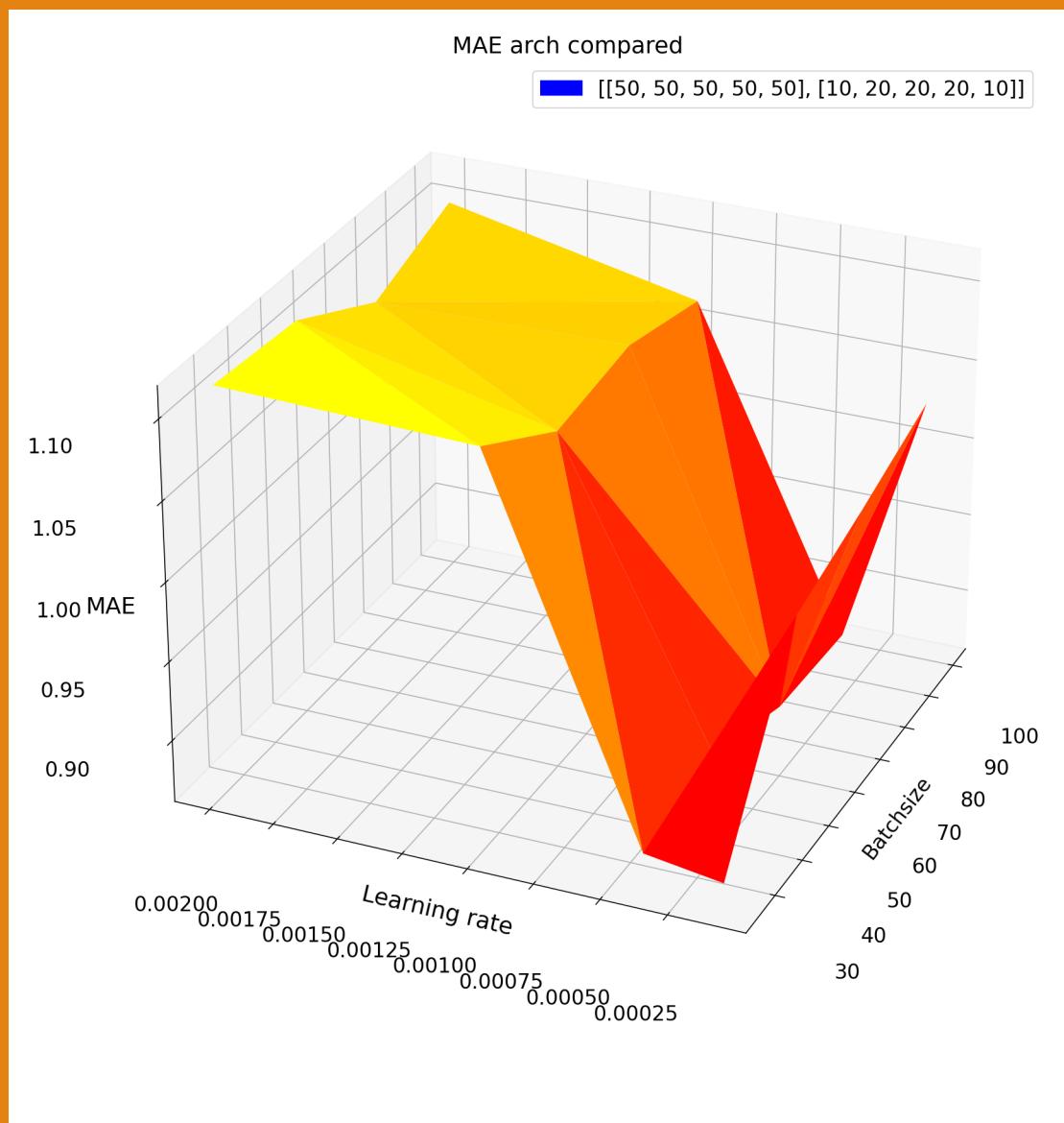
MAE arch compared

A legend box containing two entries: a blue square followed by the text "[[20, 20, 20, 20, 20], [30, 30, 30, 30, 30]]".

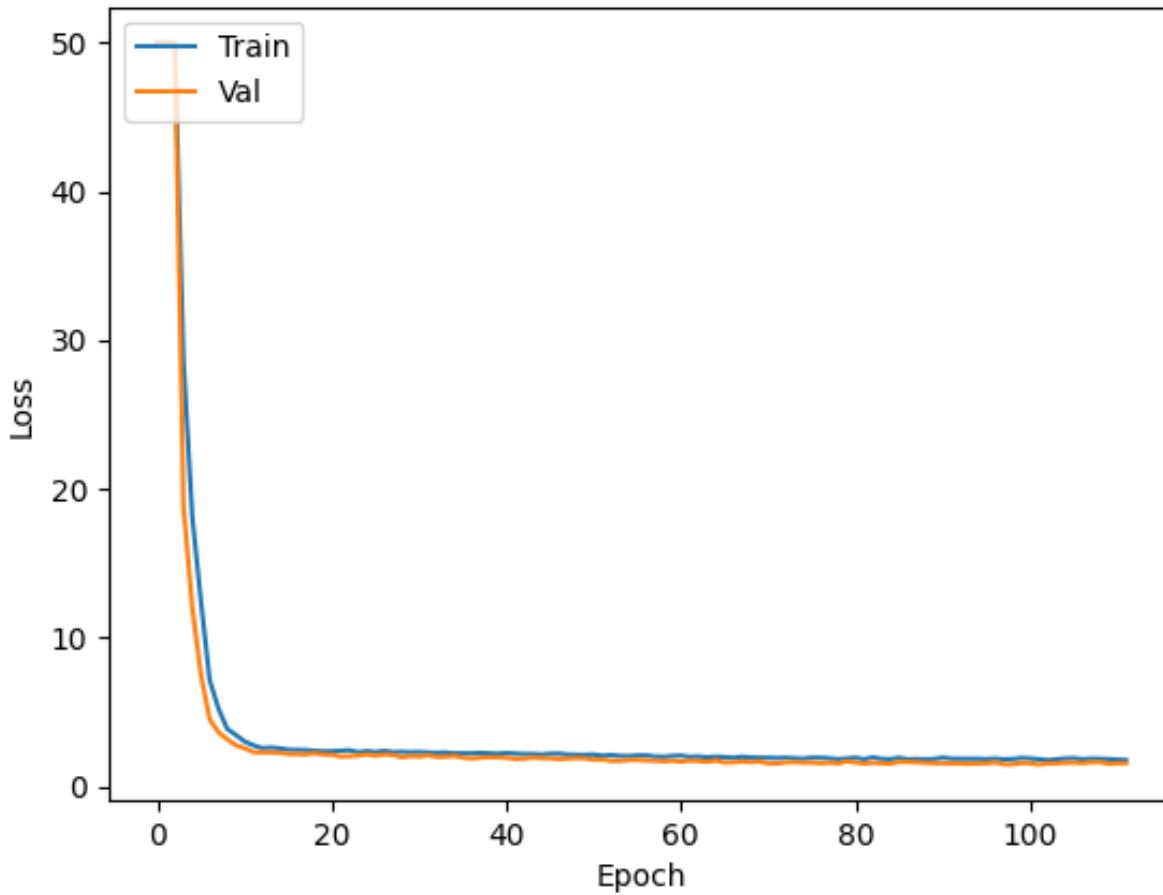


# Grid search ConvNets

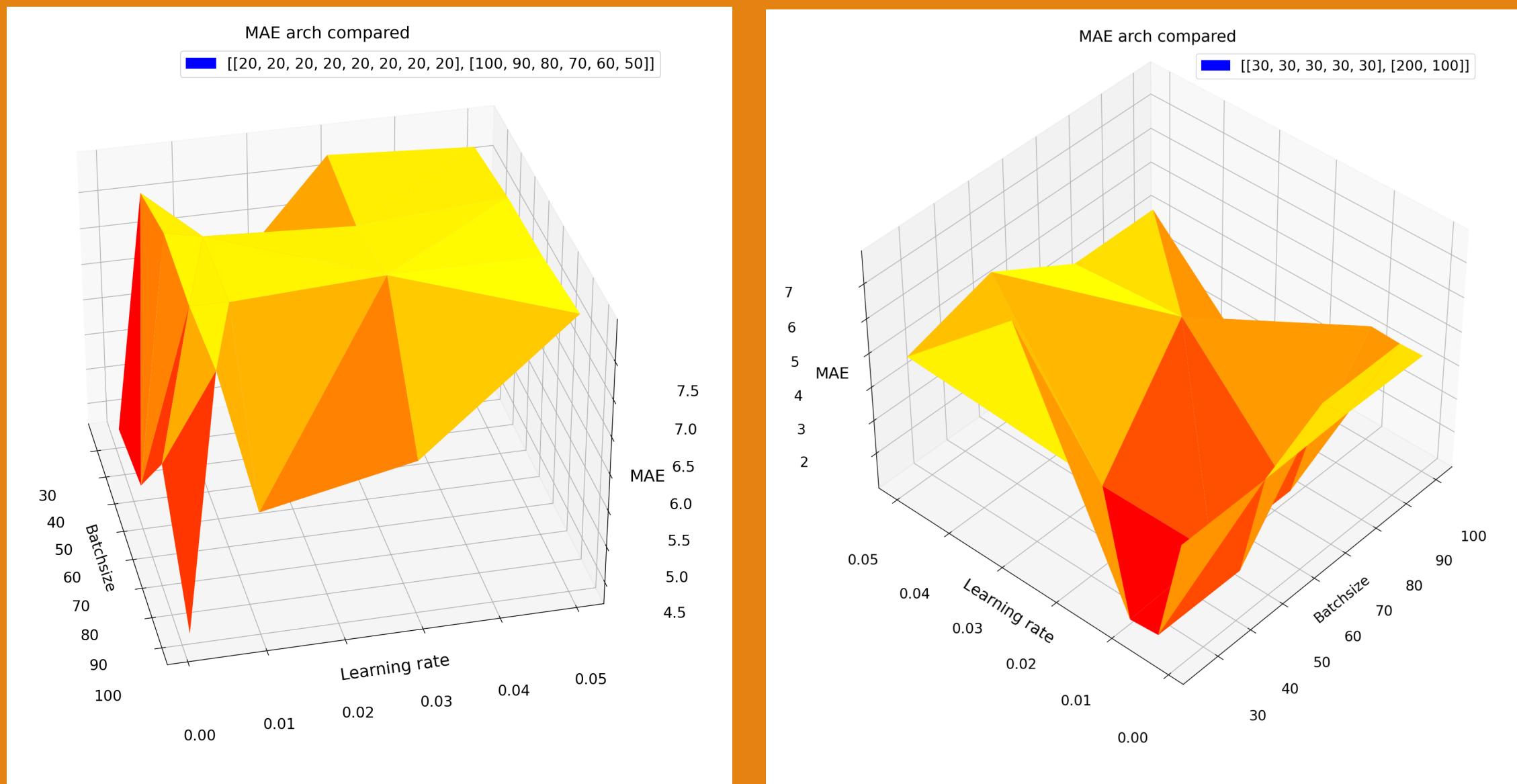
# Grid search ConvNets



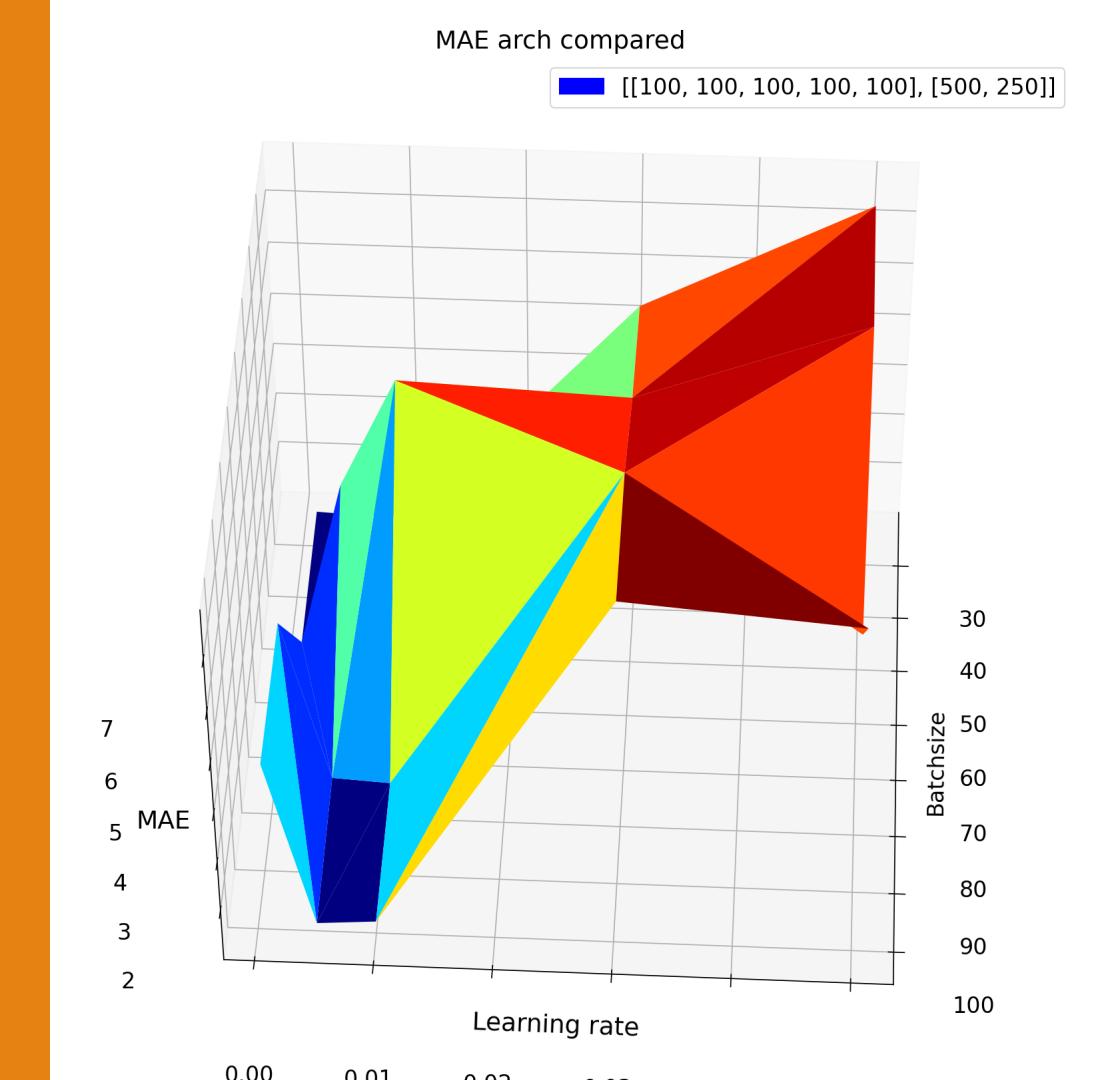
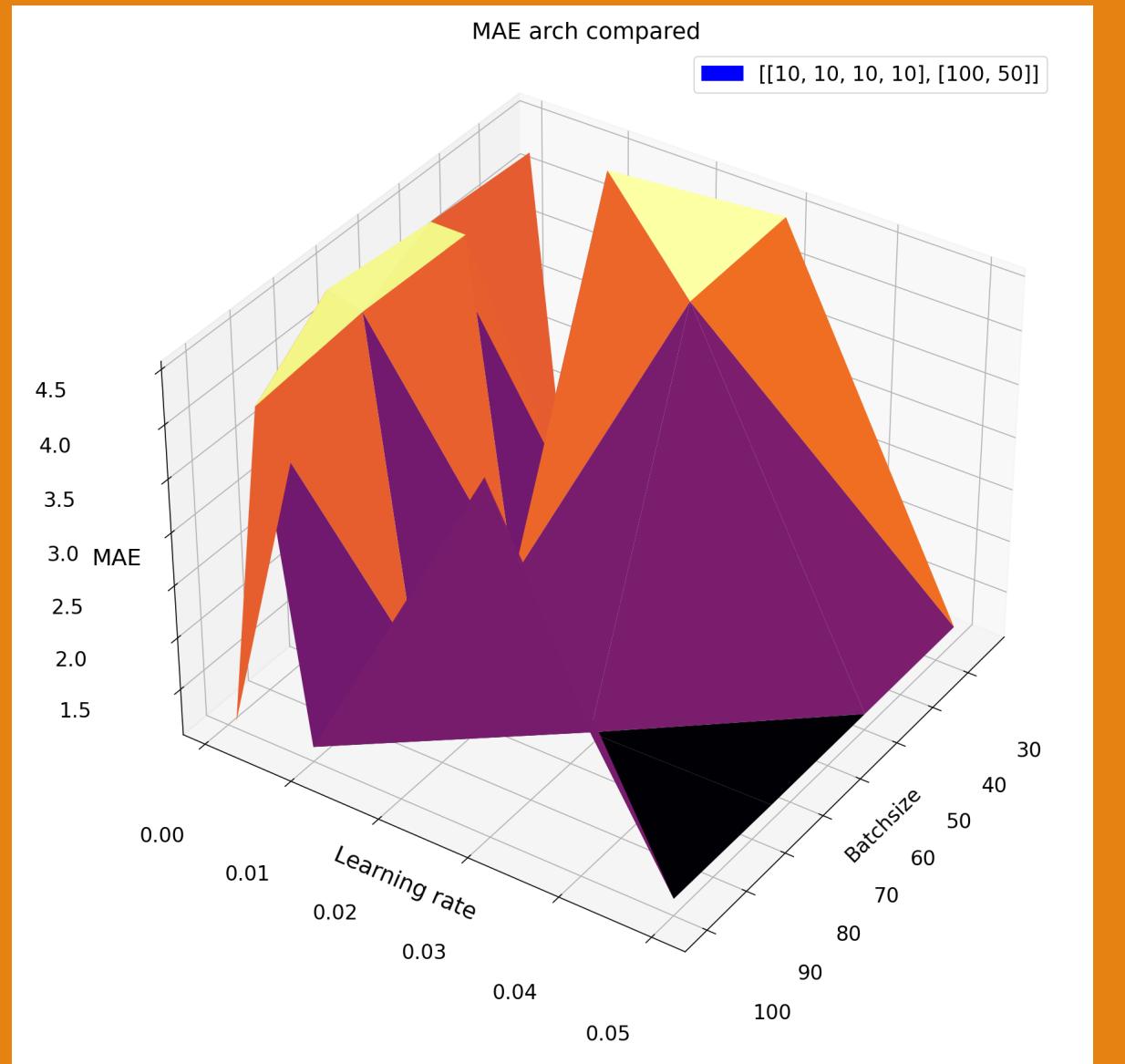
### Model Loss



ConvNets Loss-Curve

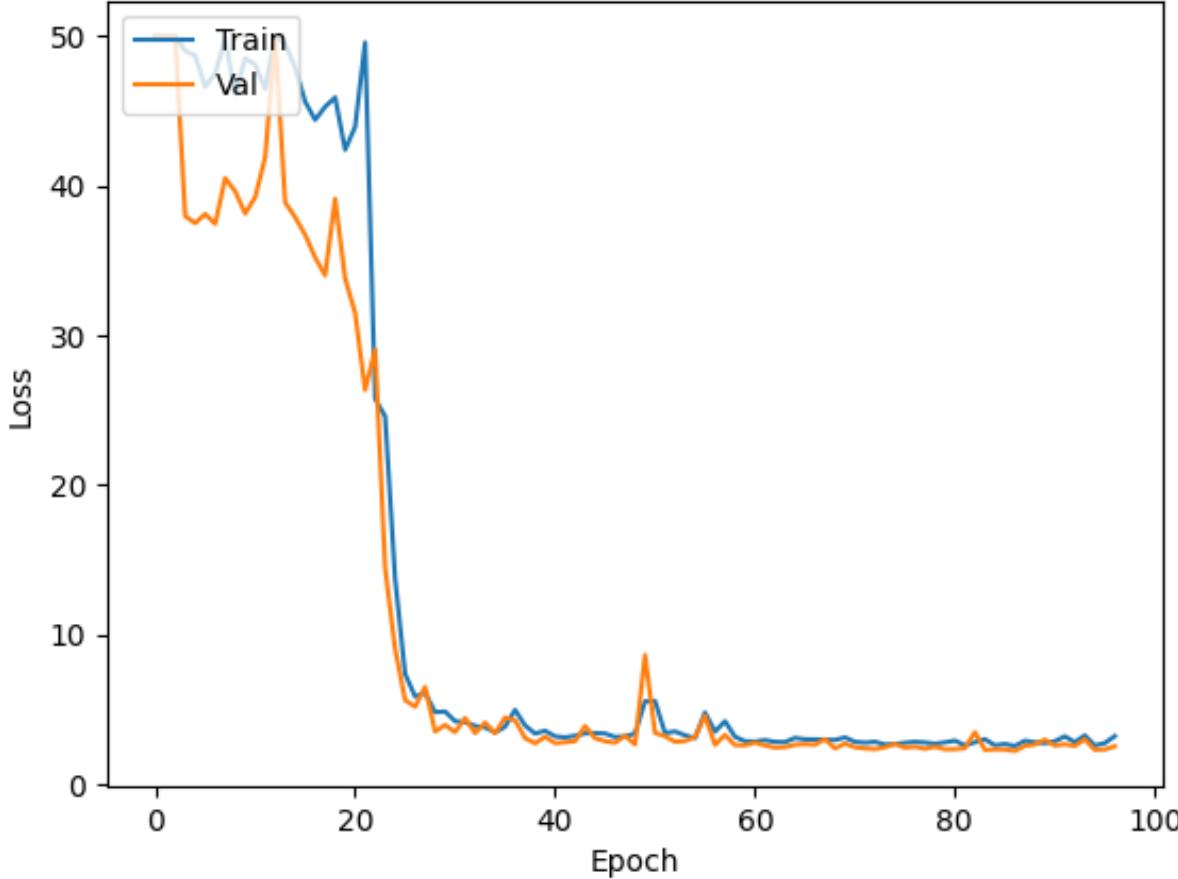


# Grid search LSTM



# Grid search LSTM

### Model Loss



## LSTM Loss-Curve

# Auswertung

---

- Training in MLPs und ConvNets sehr zuverlässig
- LSTM performt zunächst schlechter
- Das Training der LSTMs dauerte am längsten(ca. 25s/epoch)
- Schnelles Training bei ConvNets und MLP (ca. 3s/epoch)
- Besten Ergebnisse: ca. 0.86 MAE
- Die besten 137 von 199 waren besser als wir!
- Das beste mit etwa 45% des Fehlers

```
type: convs platz 1
type: mlps platz 2
type: convs platz 3
type: convs platz 4
type: convs platz 5
type: convs platz 6
type: convs platz 7
type: mlps platz 8
type: convs platz 9
type: convs platz 10
type: convs platz 11
type: convs platz 12
type: mlps platz 13
type: mlps platz 14
type: mlps platz 15
type: convs platz 16
type: mlps platz 17
type: convs platz 18
type: convs platz 19
type: mlps platz 20
type: convs platz 21
type: mlps platz 22
type: mlps platz 23
type: mlps platz 24
type: mlps platz 25
type: mlps platz 26
type: mlps platz 27
type: convs platz 28
type: convs platz 29
type: convs platz 30
type: convs platz 31
type: mlps platz 32
type: mlps platz 33
type: mlps platz 34
type: convs platz 35
type: mlps platz 36
type: convs platz 37
type: convs platz 38
type: convs platz 39
type: convs platz 40
type: mlps platz 41
type: convs platz 42
type: mlps platz 43
type: convs platz 44
type: mlps platz 45
type: convs platz 46
type: mlps platz 47
type: mlps platz 48
type: mlps platz 49
type: mlps platz 50
type: convs platz 51
type: mlps platz 52
type: mlps platz 53
type: convs platz 54
type: mlps platz 55
type: mlps platz 56
type: mlps platz 57
type: mlps platz 58
type: convs platz 59
type: mlps platz 60
type: mlps platz 61
type: convs platz 62
type: mlps platz 63
type: mlps platz 64
type: convs platz 65
type: convs platz 66
type: mlps platz 67
type: mlps platz 68
type: mlps platz 69
type: mlps platz 70
type: mlps platz 71
type: mlps platz 72
type: convs platz 73
type: mlps platz 74
type: mlps platz 75
type: convs platz 76
type: convs platz 77
type: convs platz 78
type: convs platz 79
type: convs platz 80
type: convs platz 81
type: convs platz 82
type: convs platz 83
type: convs platz 84
type: convs platz 85
type: convs platz 86
type: convs platz 87
type: mlps platz 88
type: lstms platz 89
```

# Fine Tuning

---

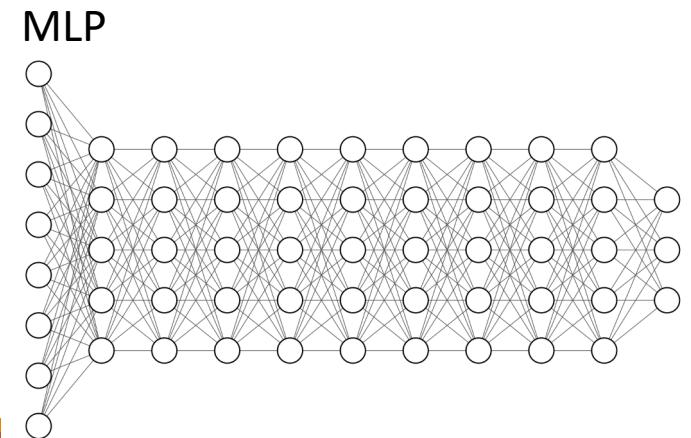
- Data Augmentation, Batch Size, Learning Rate, Patience, Architecture, Normalization, Dropout
- Data Augmentation (\*[0.5,2] , +[-10,10])
- Patience
- Dropout
- Normalization schwer zu vergleichen

# Fine Tuning Ergebnisse

---

1. MLP Layers: [500,500,500,500,500,500,500], Learning Rate: 0.00008, Augmentation, Patience: 40, Batch Size: 300, MAE 0.86 -> 0.83
2. 1D ConvNet Kernels:[10,10,10,10], Kernel Sizes:[5,5,5,5], Learning Rate: 0.00005, Augmentation, Patience: 40, Batch Size: 300, MAE : 0.86 → 0.8365

! statt global average pooling lieber flatten
3. LSTM Layers: [30,30,30,30,30],[200,100], Learning Rate: 0.01, Augmentation, Patience: 40, Batch Size: 100, MAE 1.1 → 0.88
4. Transformer Transformerblocks: 2 , Learning Rate: 0.0001,  
Batch size: 4 (Speicherbedingt), MLP: [128,128,128]  
MAE : 1.0893

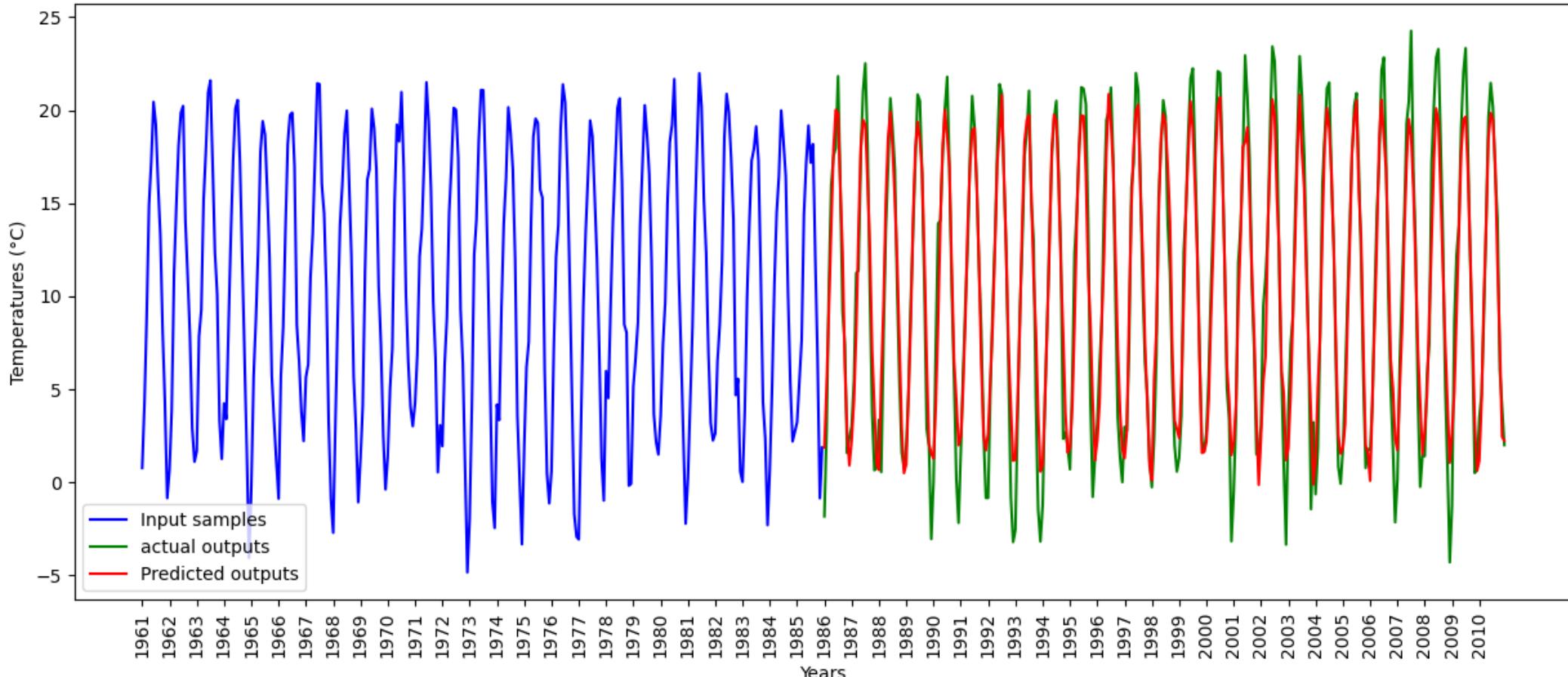


# Predictions

Beste Transformer:

(bekommt 840 (70 Jahre) Monate und predicted 300 (25 Jahre))

Predicted Temperatures for the next 25 years in ÇorumTurkey

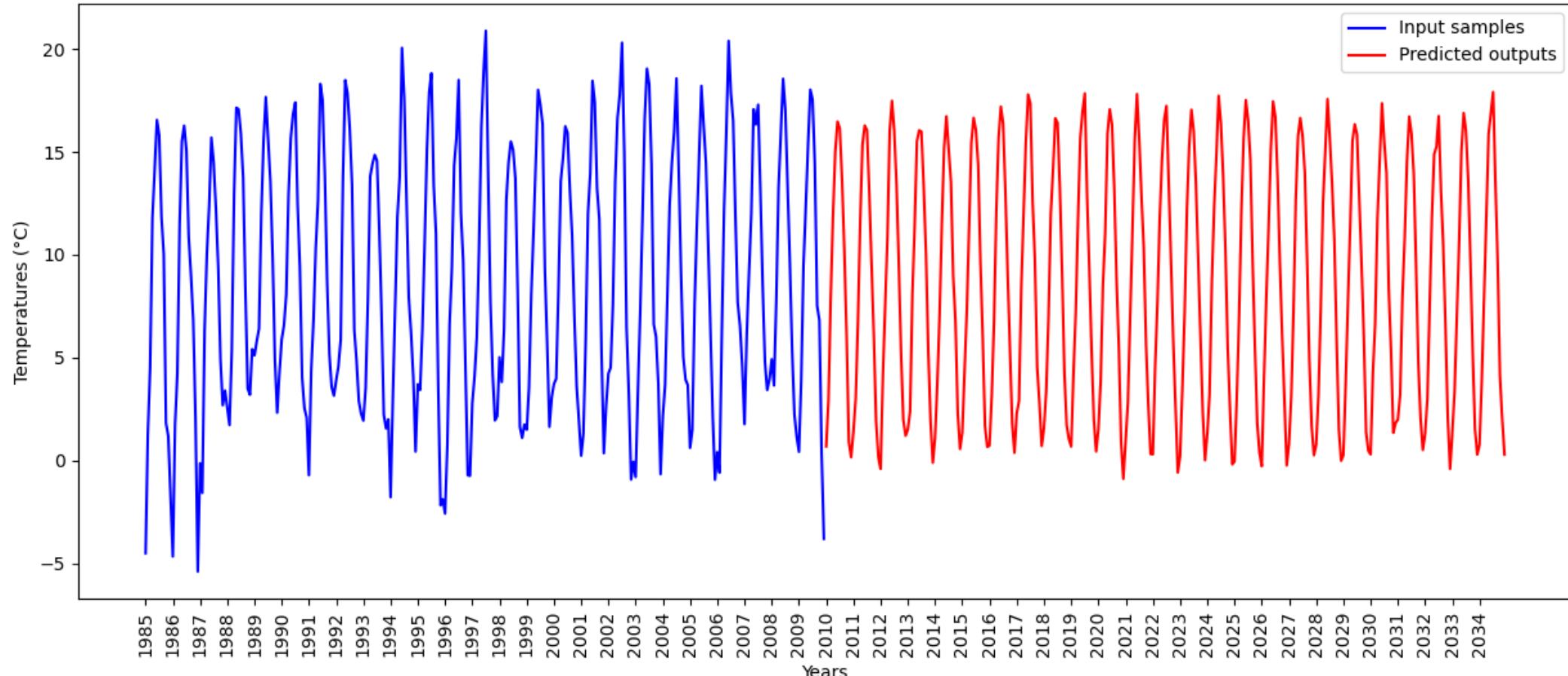


# Predictions

Beste Transformer:

(bekommt 840 (70 Jahre) Monate und predicted unbekannte 300 (25 Jahre))

Predicted Temperatures for the next 25 years in ÅrhusDenmark





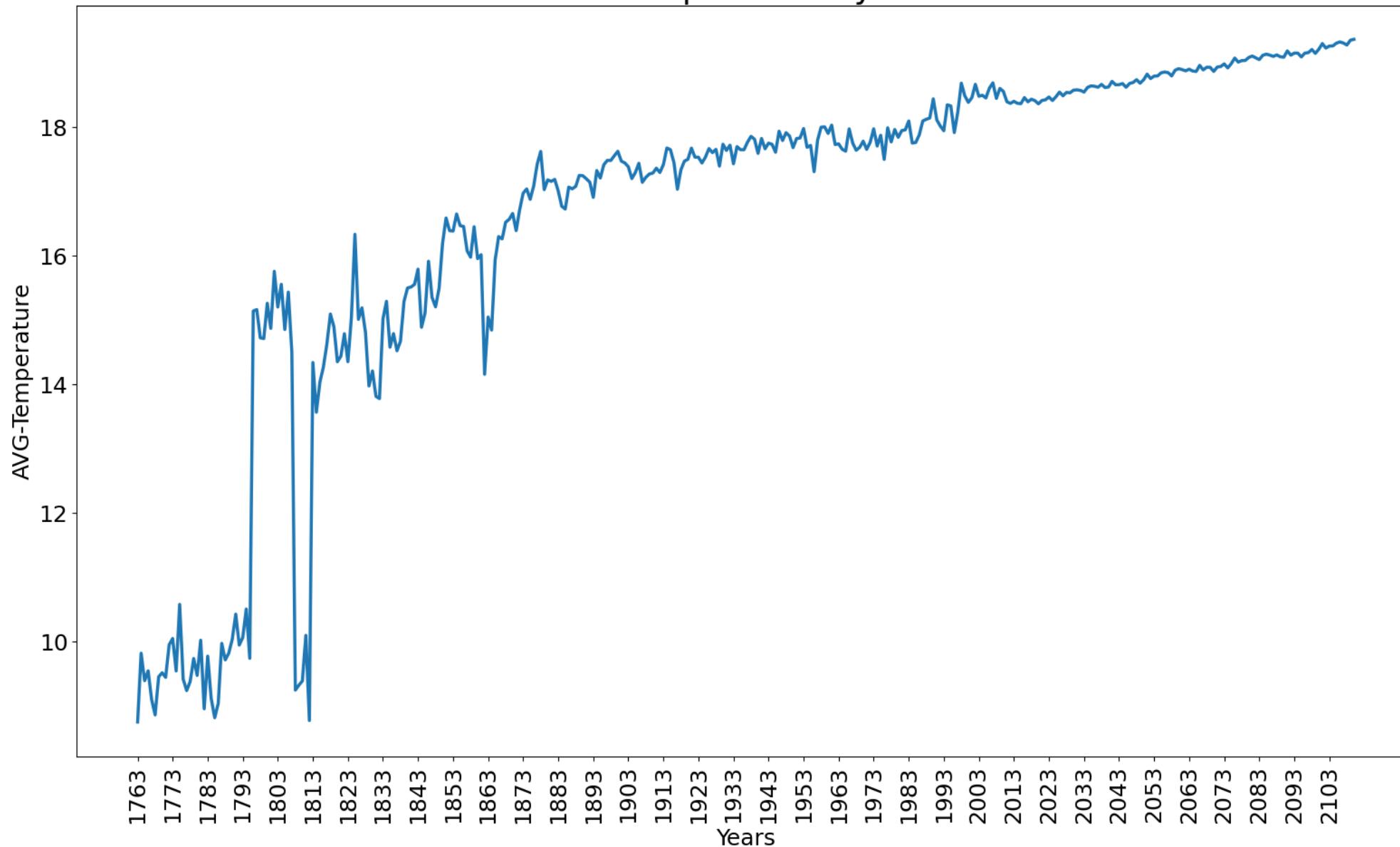
# Klimavorhersagen

---

Ungefähr seit Mitte des 20 Jahrhunderts menschlicher Einfluss deutlicher sichtbar

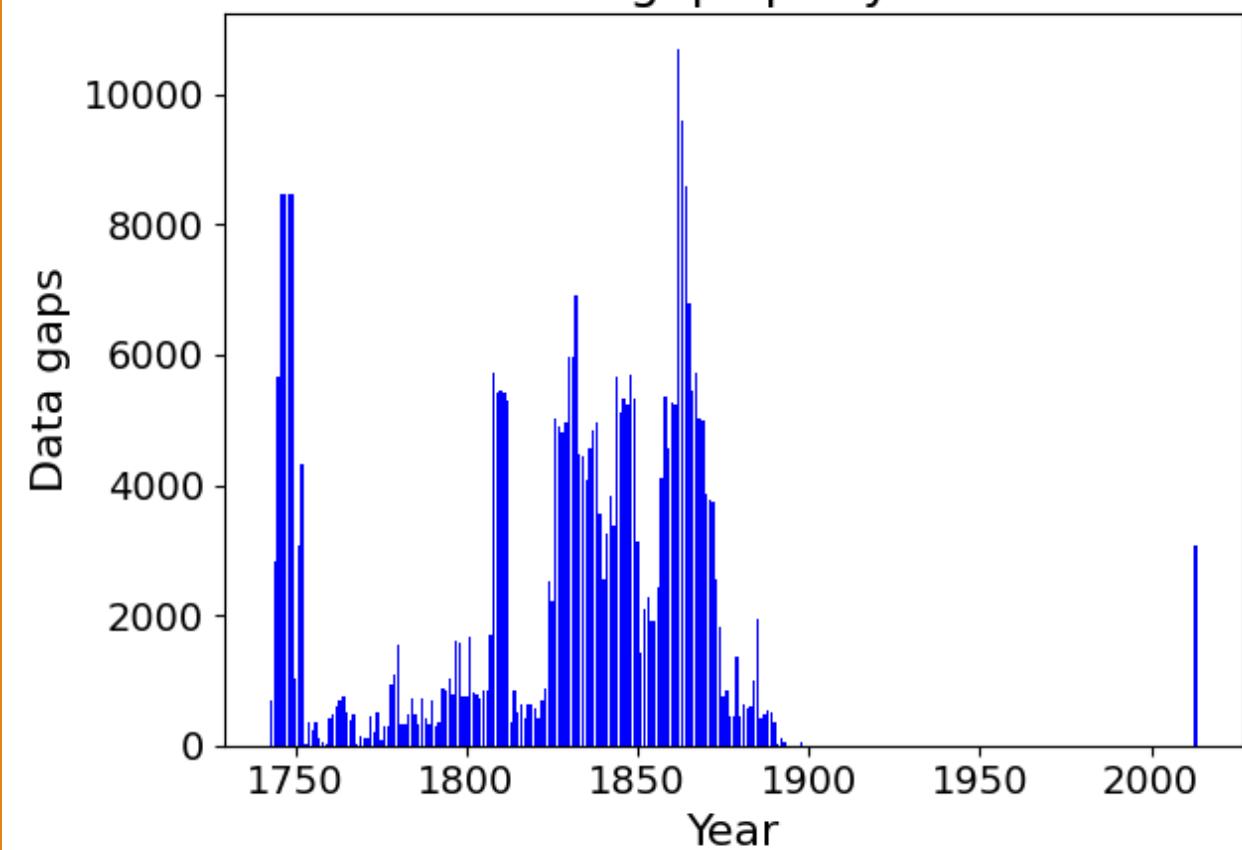
- Vorhersagen
- Aussagekraft

## AVG-Temperature by Year

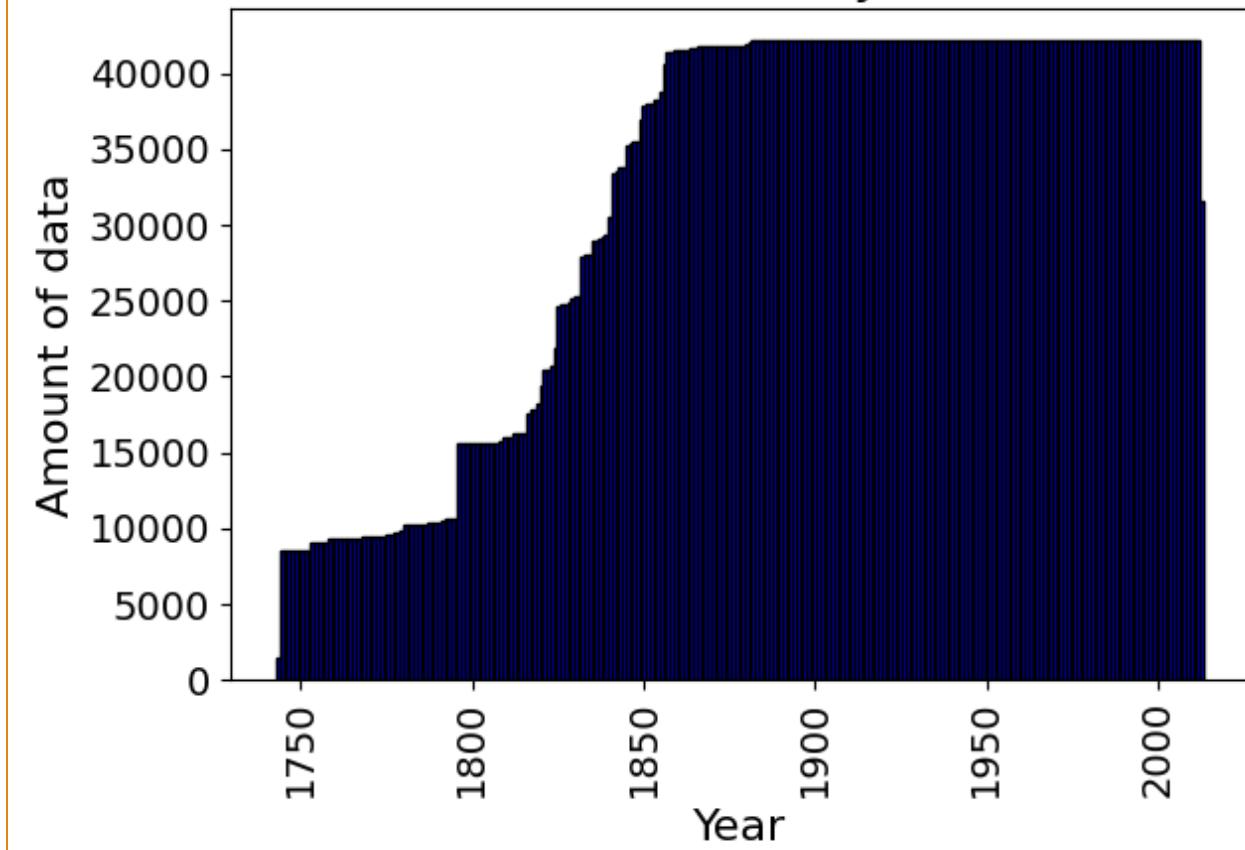


Geplottet nur da, wo es 840 Monate (70 Jahre) konstant gab

### Data gaps per year



### Data Amount by Year



Klimavorhersagen