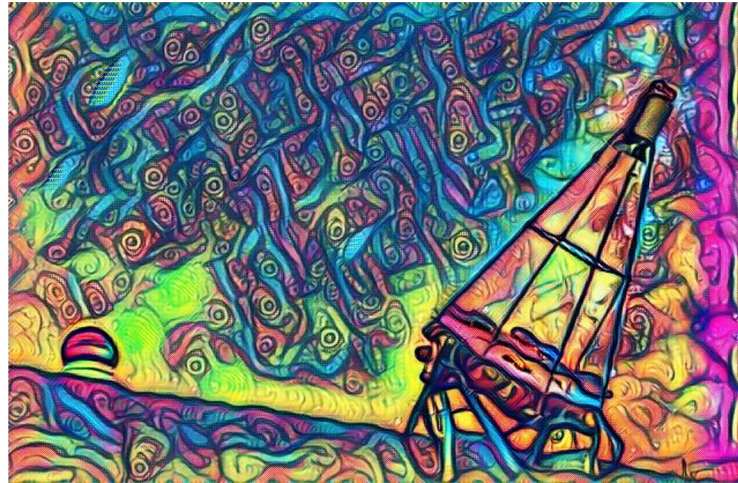


---

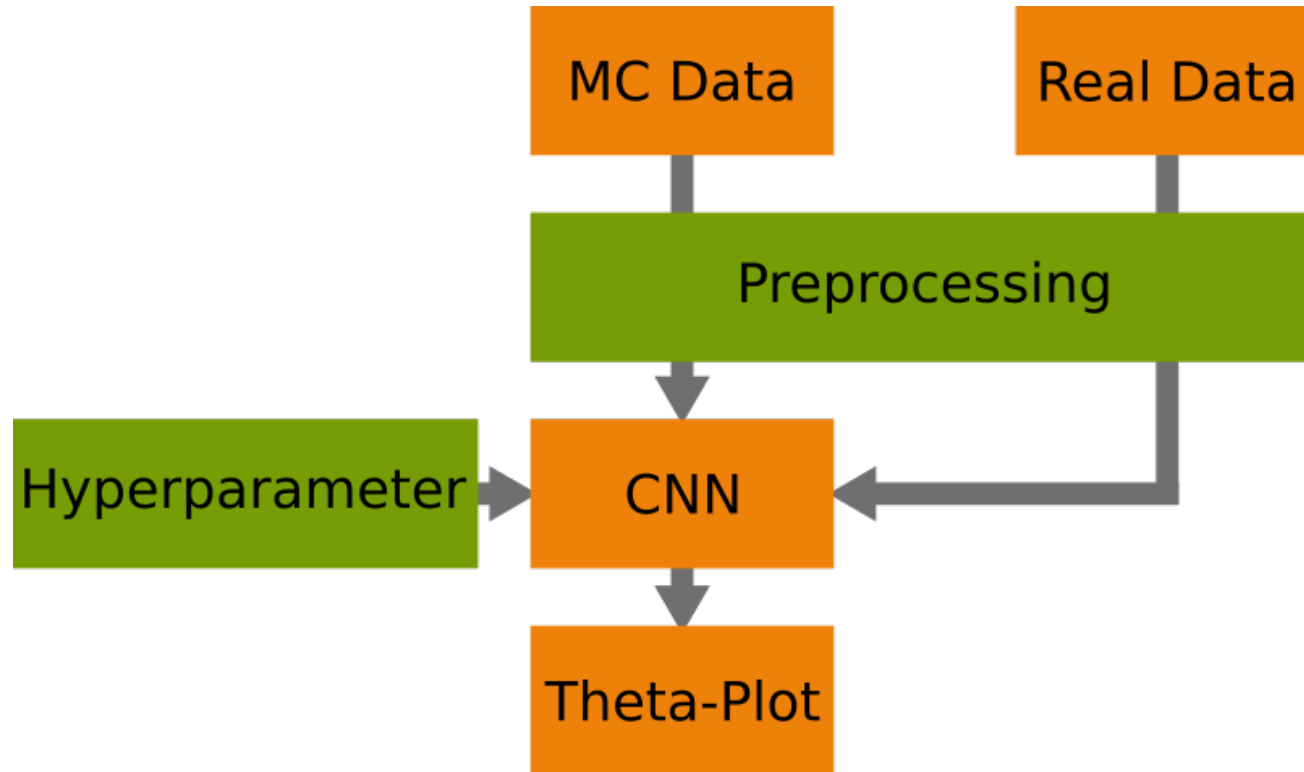
# Gamma-Hadron Separation with Deep Learning for the First G-APD Cherenkov Telescope

Bachelorvortrag – Jan Moritz Behnken



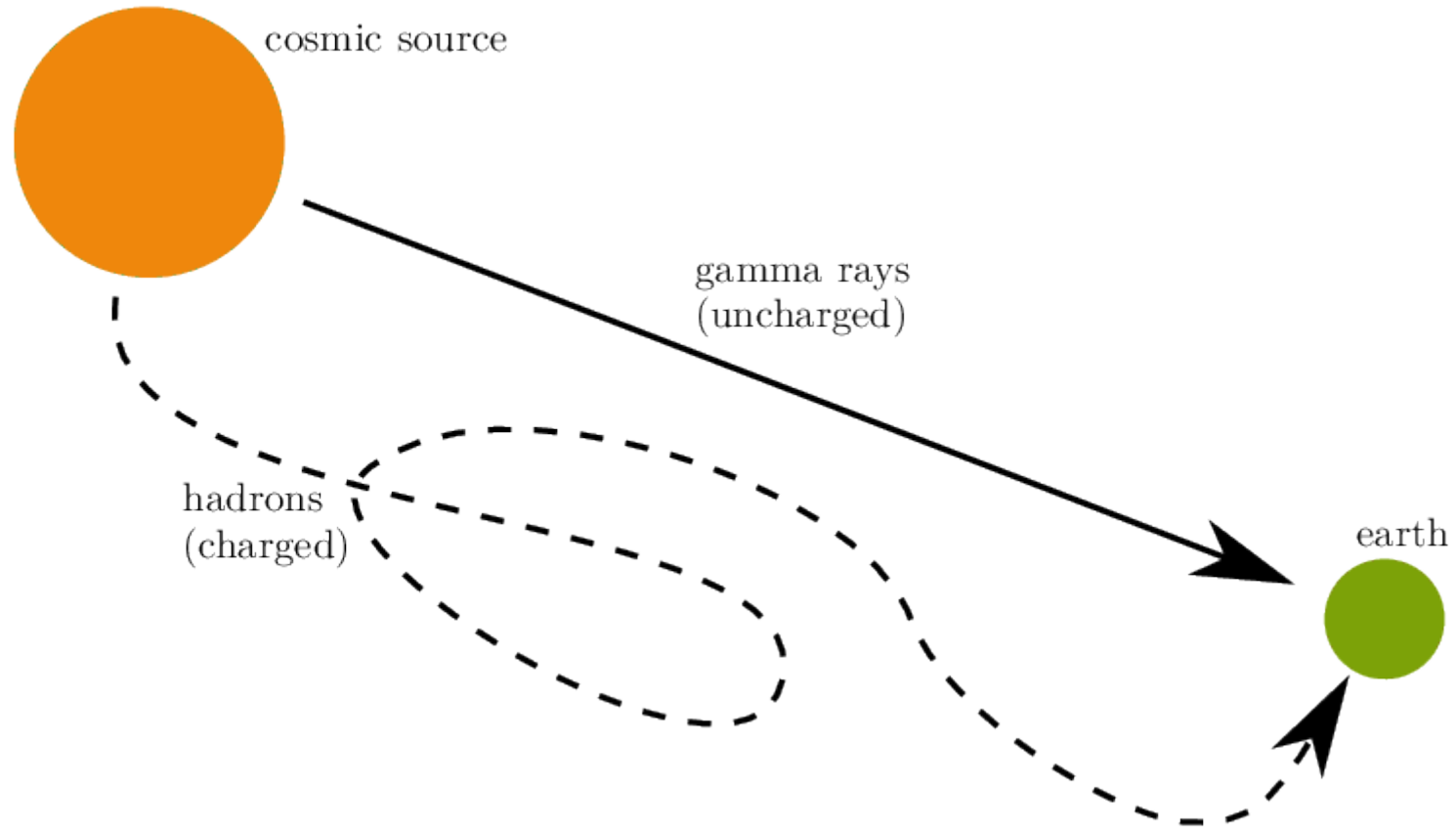
# Gamma-Hadron Separation mit CNNs bei FACT

---

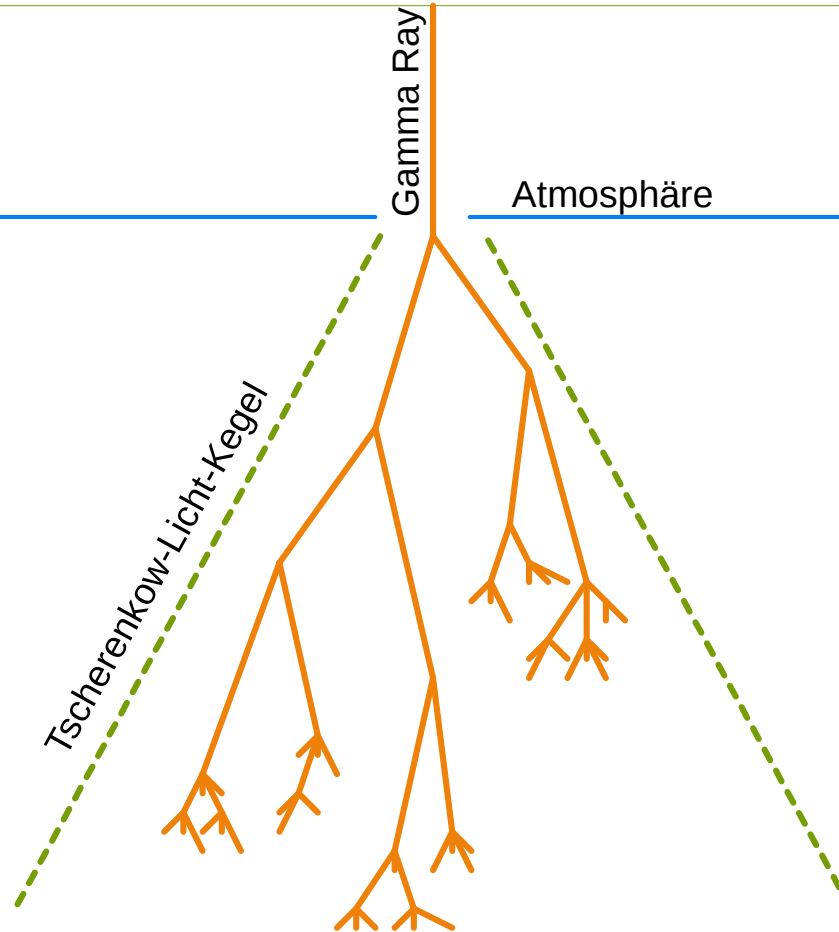


# Gamma Rays breiten sich geradlinig im Kosmos aus

---



# Luftschauer kosmischer Strahlung emittieren Tscherenkow-Licht

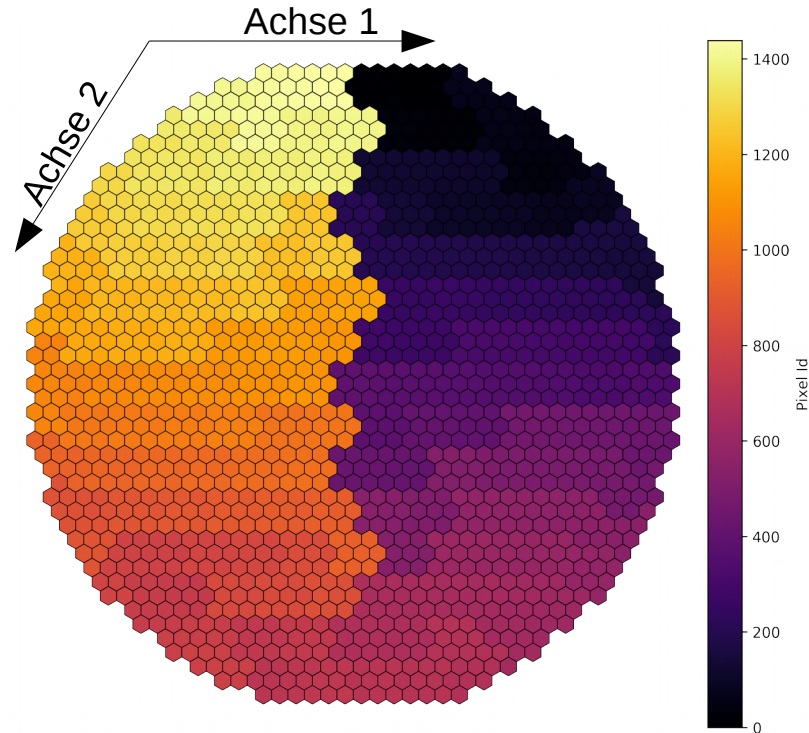


# First G-APD Cherenkov Telescope (FACT)

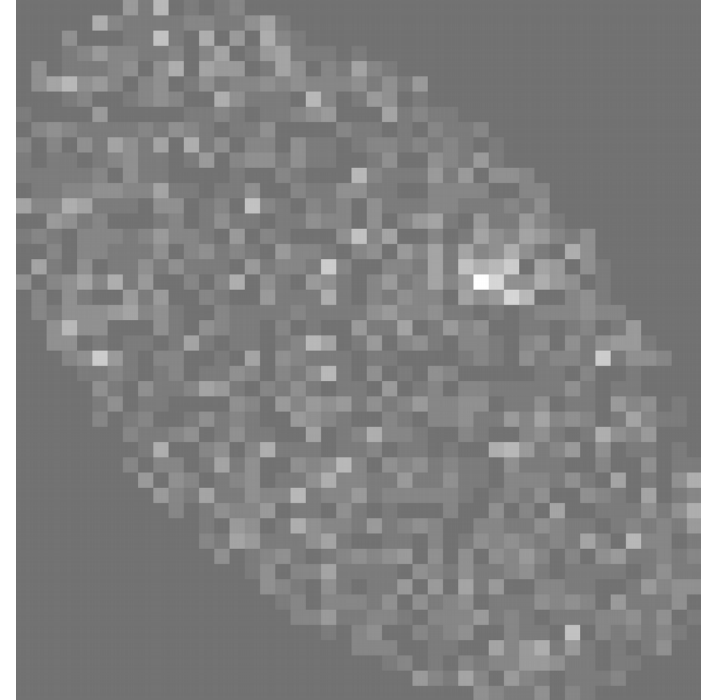
---



# Transformation von hexagonalen zu quadratischen Pixeln



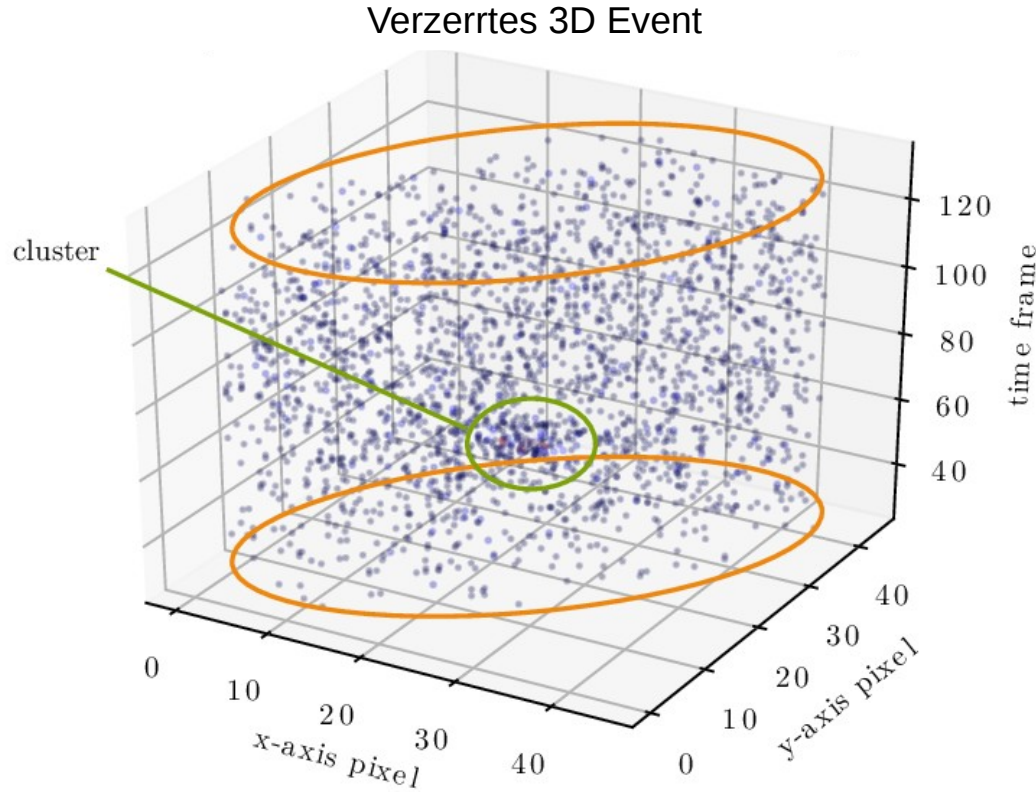
FACT Kamerabild



Standardisiertes Input-Bild



# Zeitreihen-Summation jedes Pixels reduziert die Komplexität

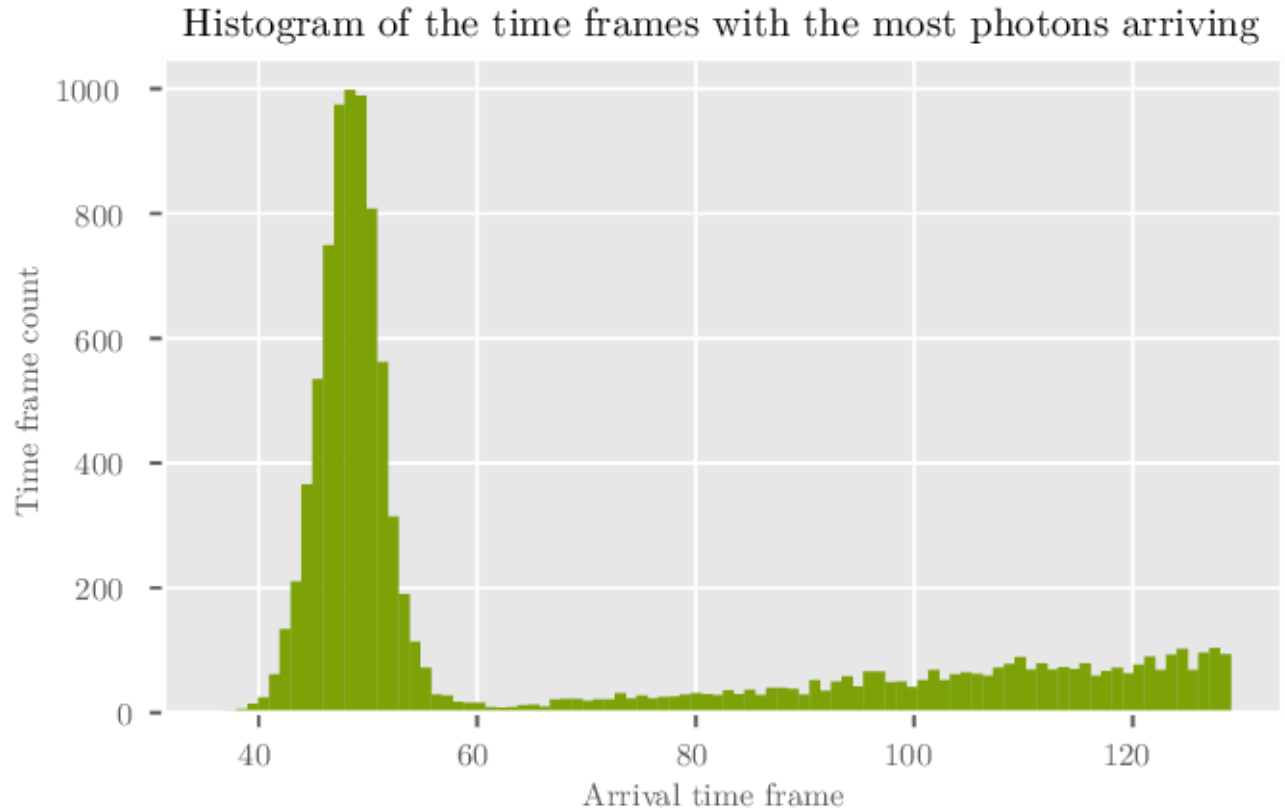


## Aufbau der Datenstruktur

- 100 aufeinanderfolgende Zeitebenen
- Pixel enthalten Zählraten der Photonen
- Zeitebenen aufsummieren reduziert 144.000 Variablen auf 1.440

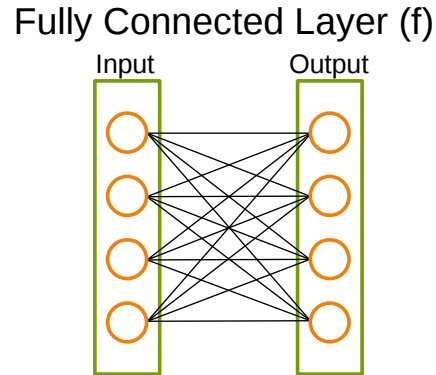
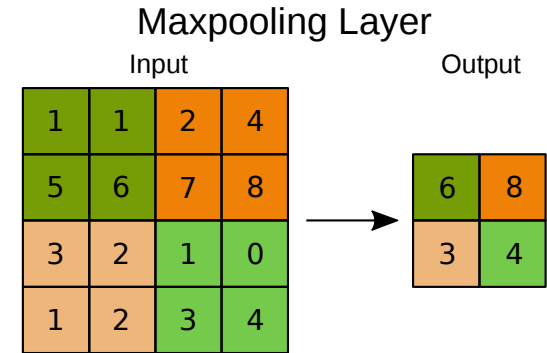
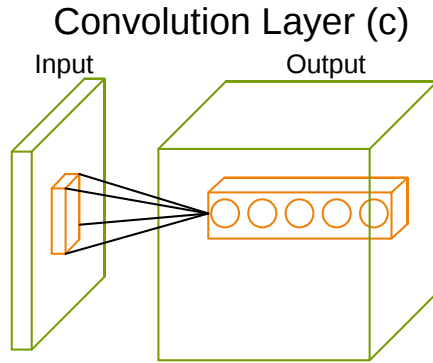
# Herausschneiden des Events verringert das Bildrauschen

- Fester Trigger-Zeitpunkt bei FACT
- Histogramm der hellsten Zeitebenen von 10.000 Events
- Herausschneiden des Peaks entfernt Bildrauschen und erhält das Signal



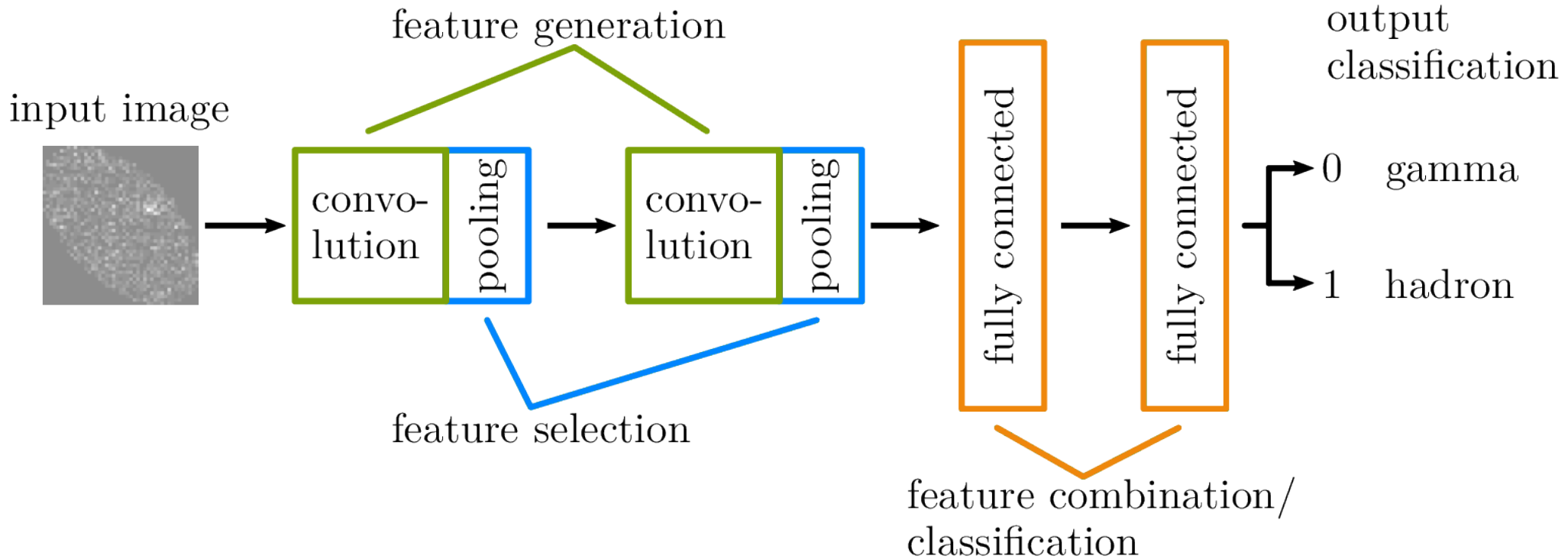


# CNNs enthalten Convolution-, Pooling- und Connected Layers



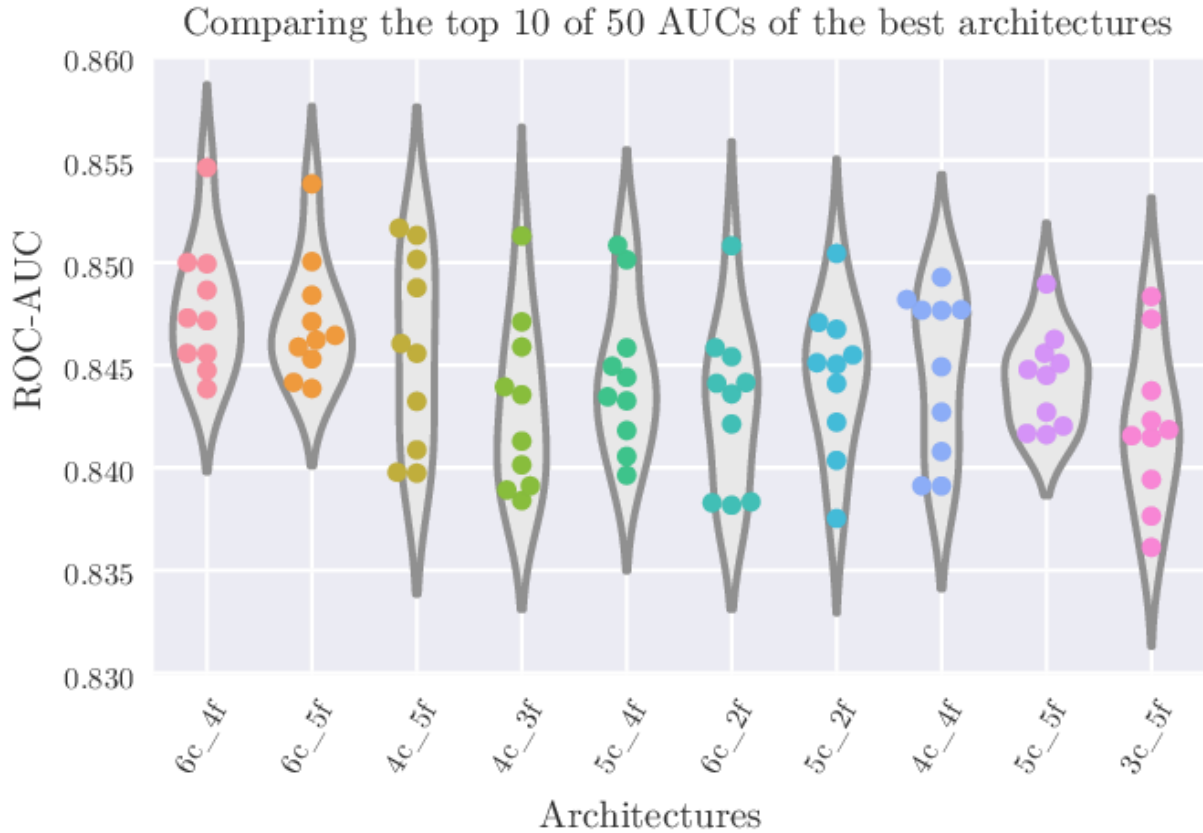
- **Convolution Layers:** Trainieren kleine neurale Netze mit Weight-Sharing um Features zu berechnen
- **Maxpooling Layers:** Reduzieren Parameter indem sie die wichtigsten Features auswählen.
- **Fully Connected Layers:** Kombinieren die Features und klassifizieren

# Beispiel CNN einer 2c\_2f Architektur

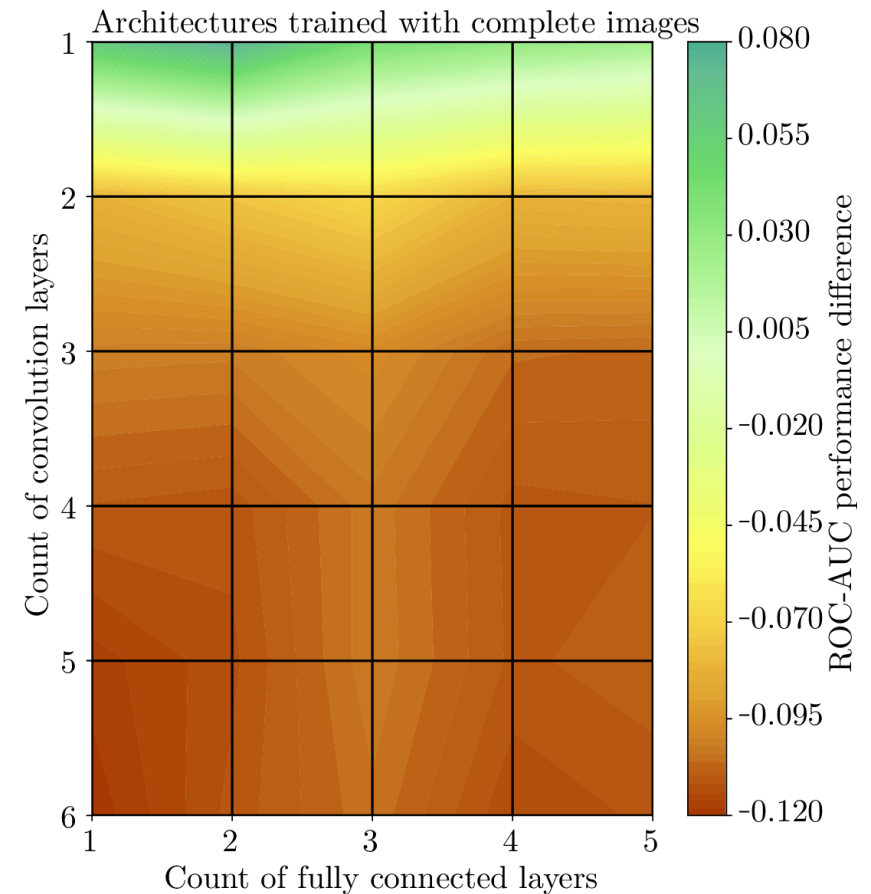
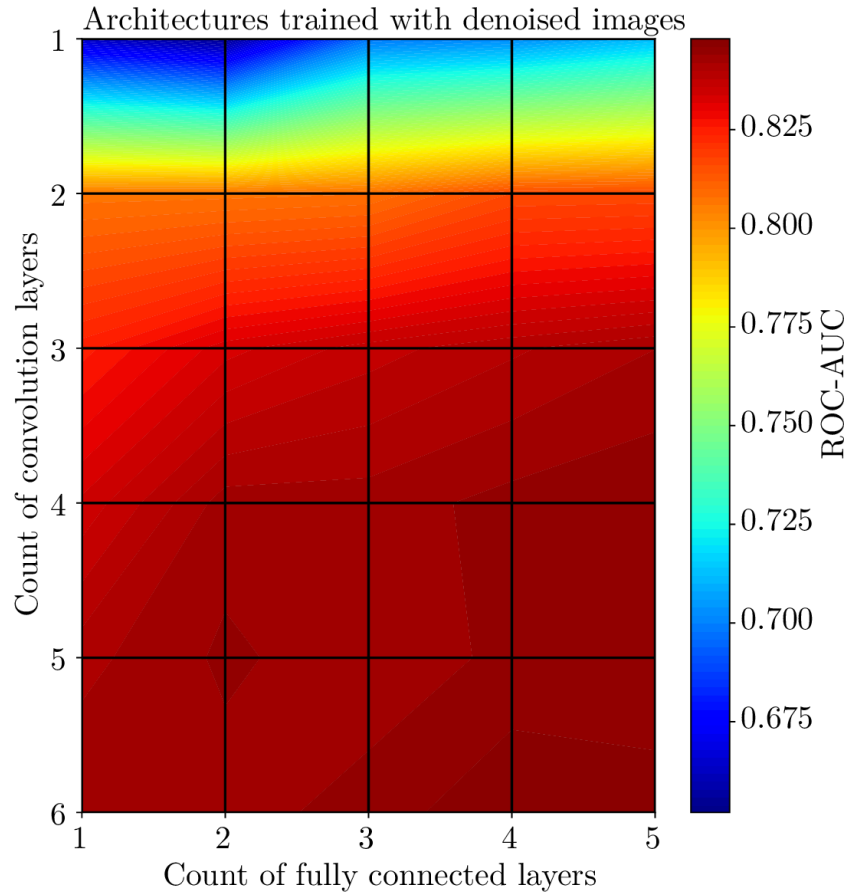


# Beste Ergebnisse mit 6 Convolution und 4 Connected Layers

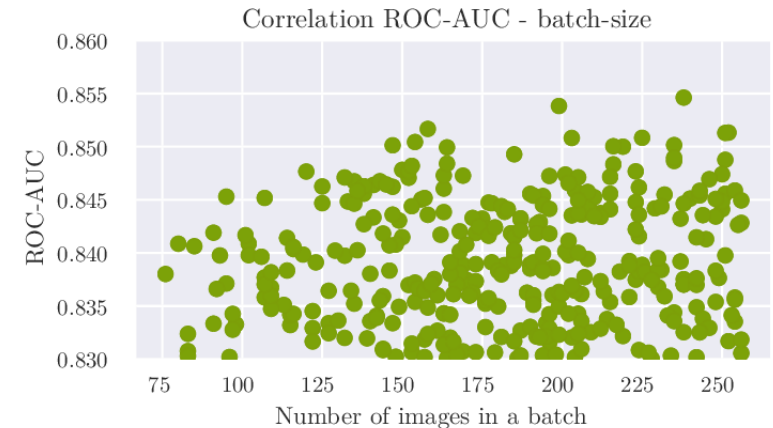
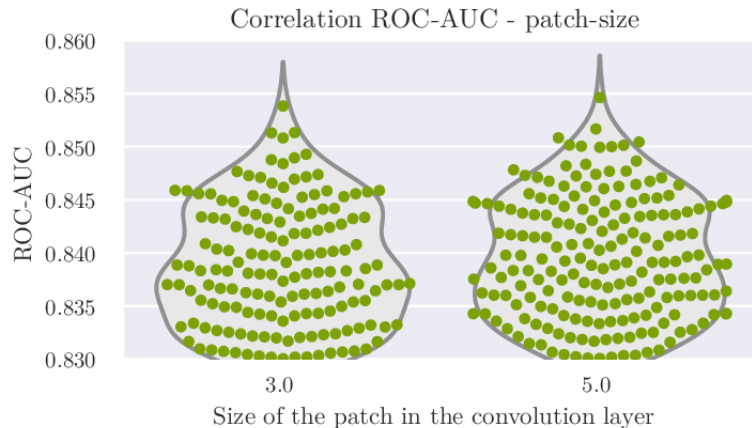
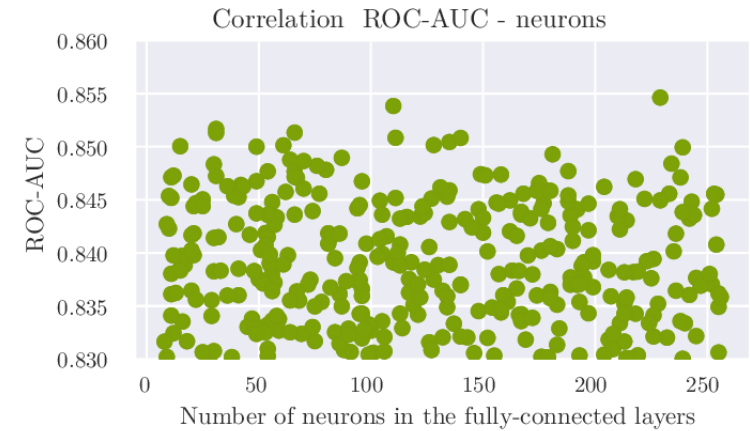
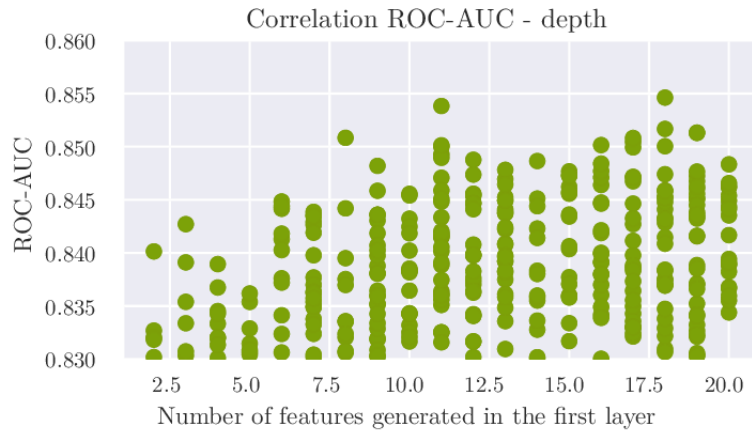
- Vergleich von 30 Netzwerk Architekturen
- 50 randomisierte Hyperparameter Grid-Searches pro Architektur



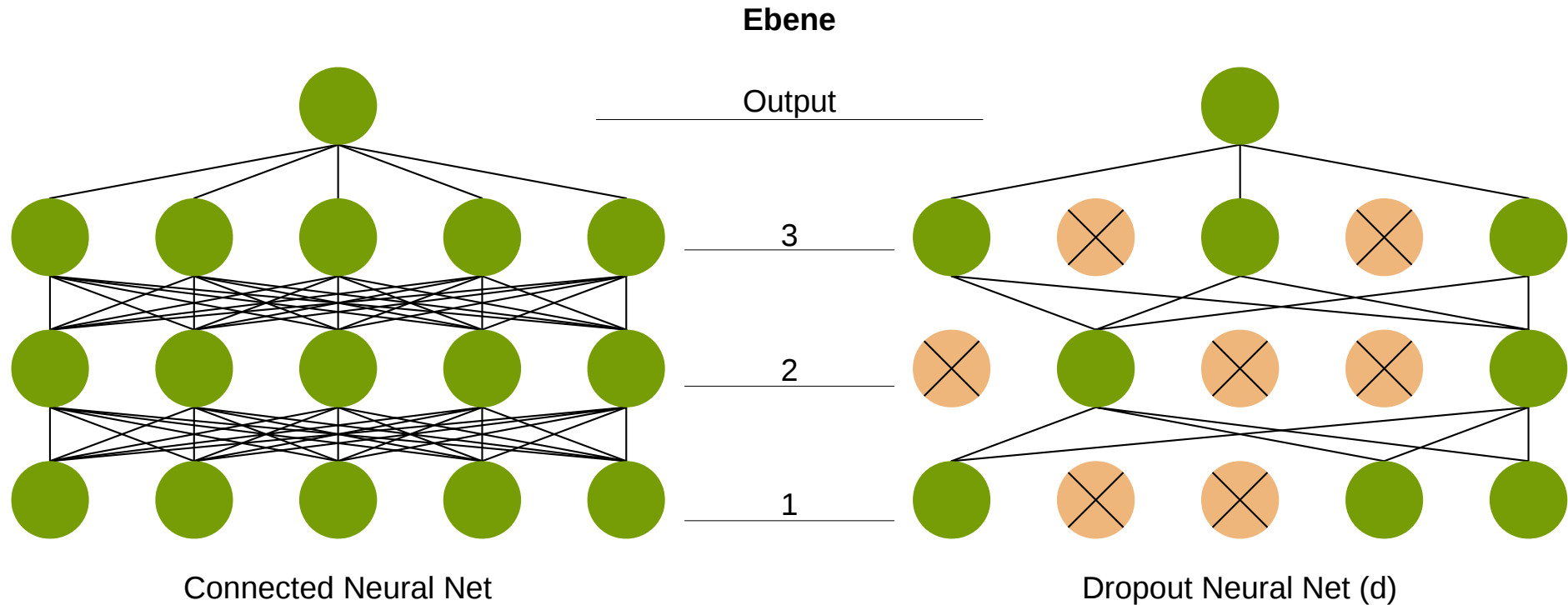
# Tiefe Netze mit entrauschten Bildern liefern die besten Ergebnisse



# Gute CNN Hyperparameter lassen sich nicht eindeutig bestimmen

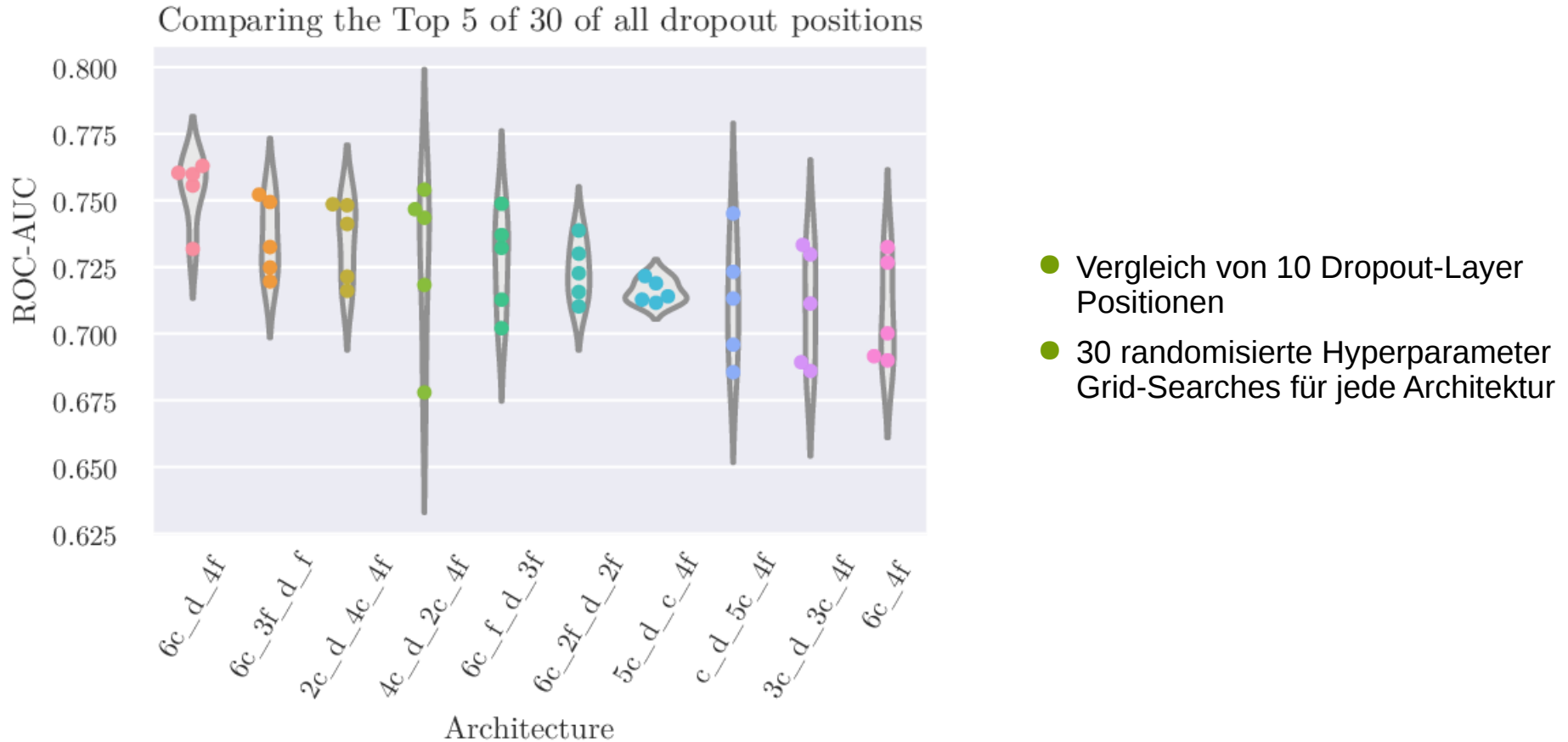


# Neuronen mit Dropout entfernen, regularisiert das Overfitting

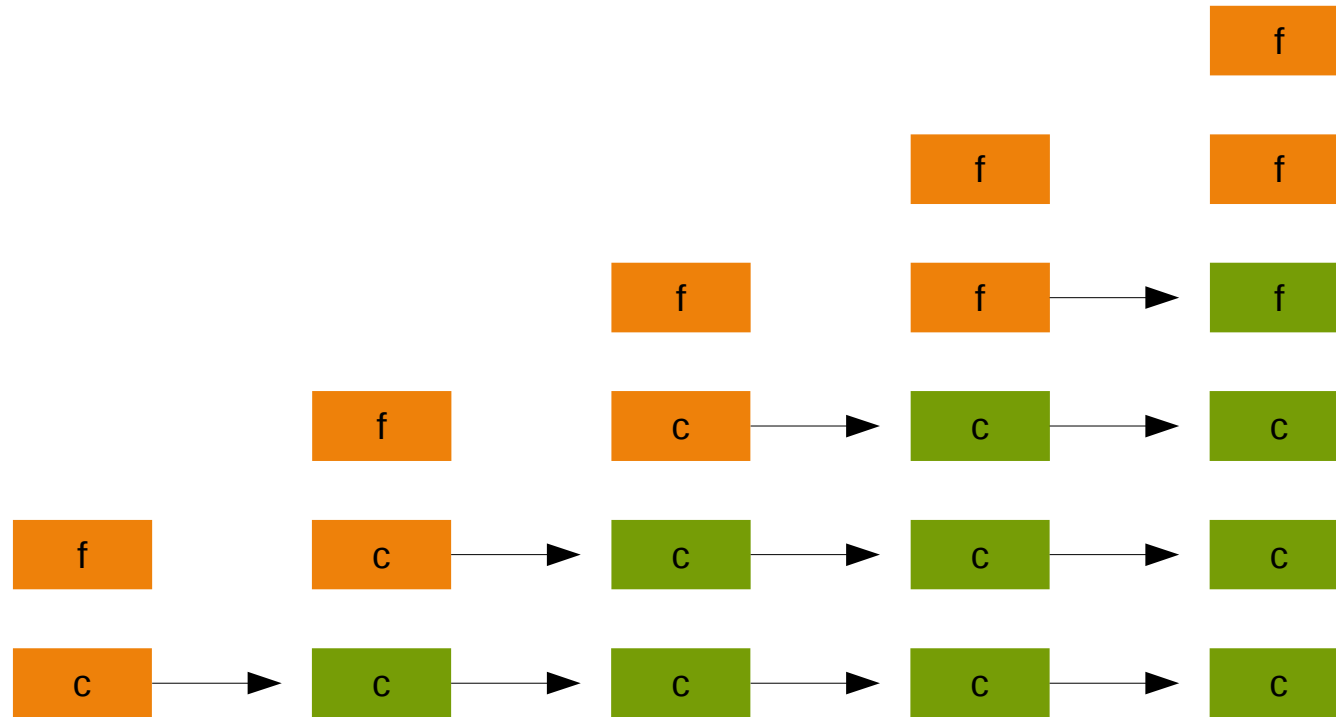




# Dropout wirkt gut zwischen Convolution und Connected Layer



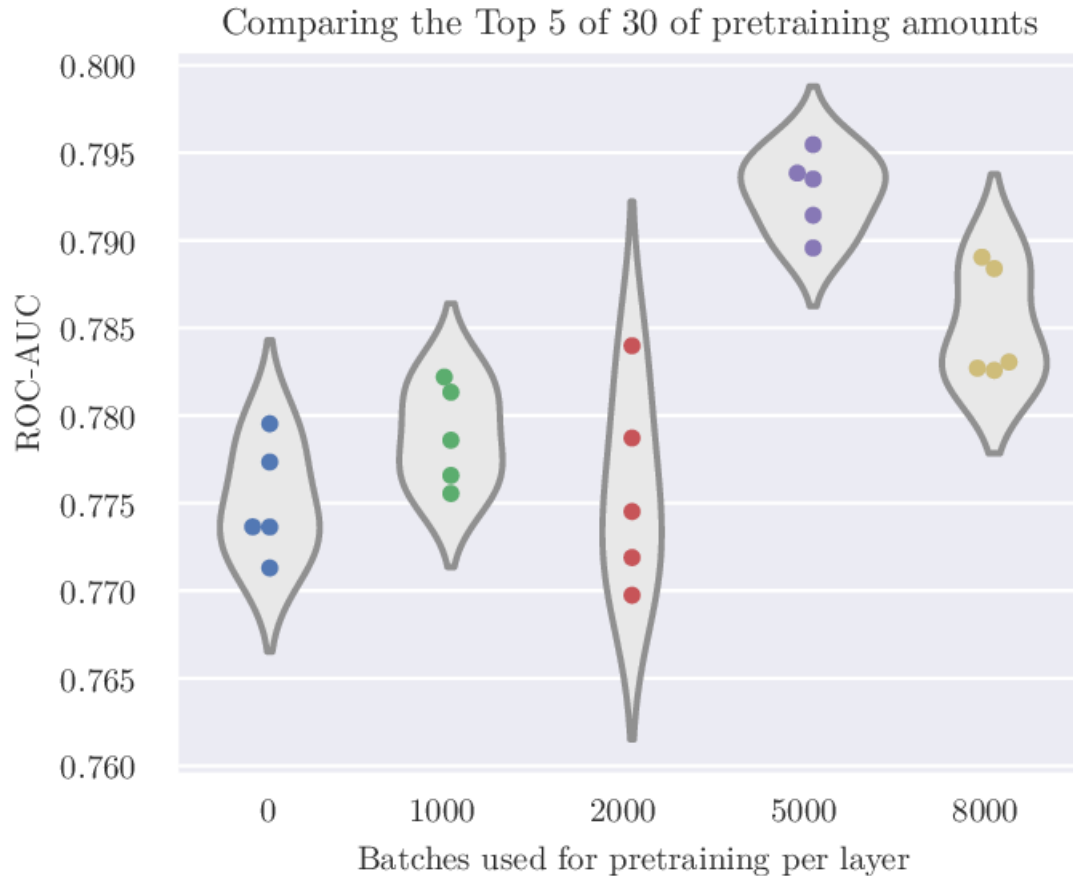
# Pretraining Layer ermöglicht die Konvergenz tiefer Netzwerke



Architekturen während des Pretraining-Prozesses

● Neu initialisierte Weights ● Pretrained Weights

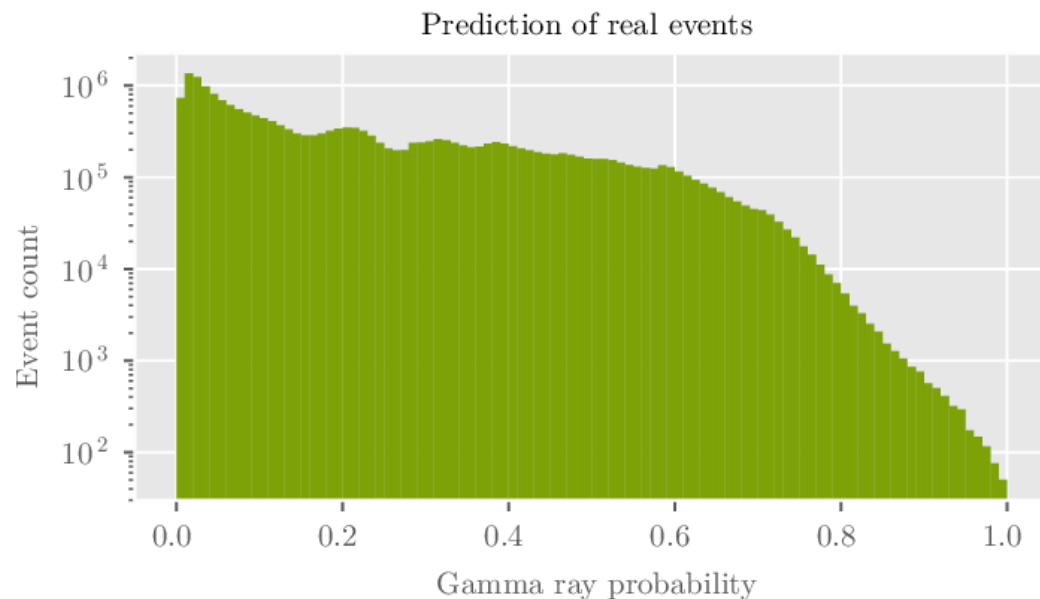
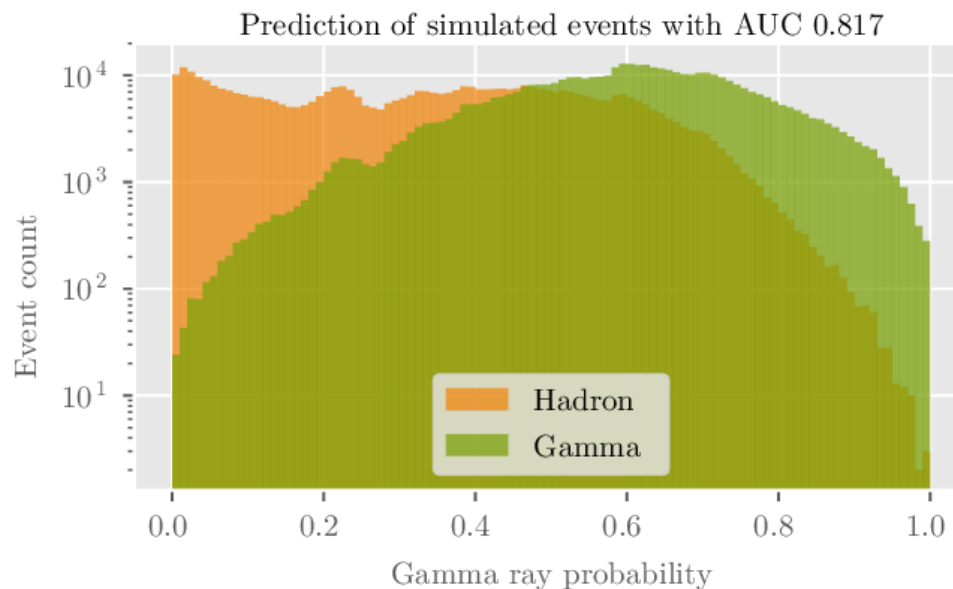
# Kurzes Pretraining hat einen positiven Effekt auf die Performance



- Vergleich von 5 Pretraining-Architekturen eines 6c\_4f Netzwerkes
- 30 randomisierte Hyperparameter Grid-Searches für jede Architektur
- Dropoutraten; c: 0.9, c-f: 0.75, f: 0.5
- Kurzes Pretraining kann die Performance erhöhen; zu langes Pretraining hemmt die Performance

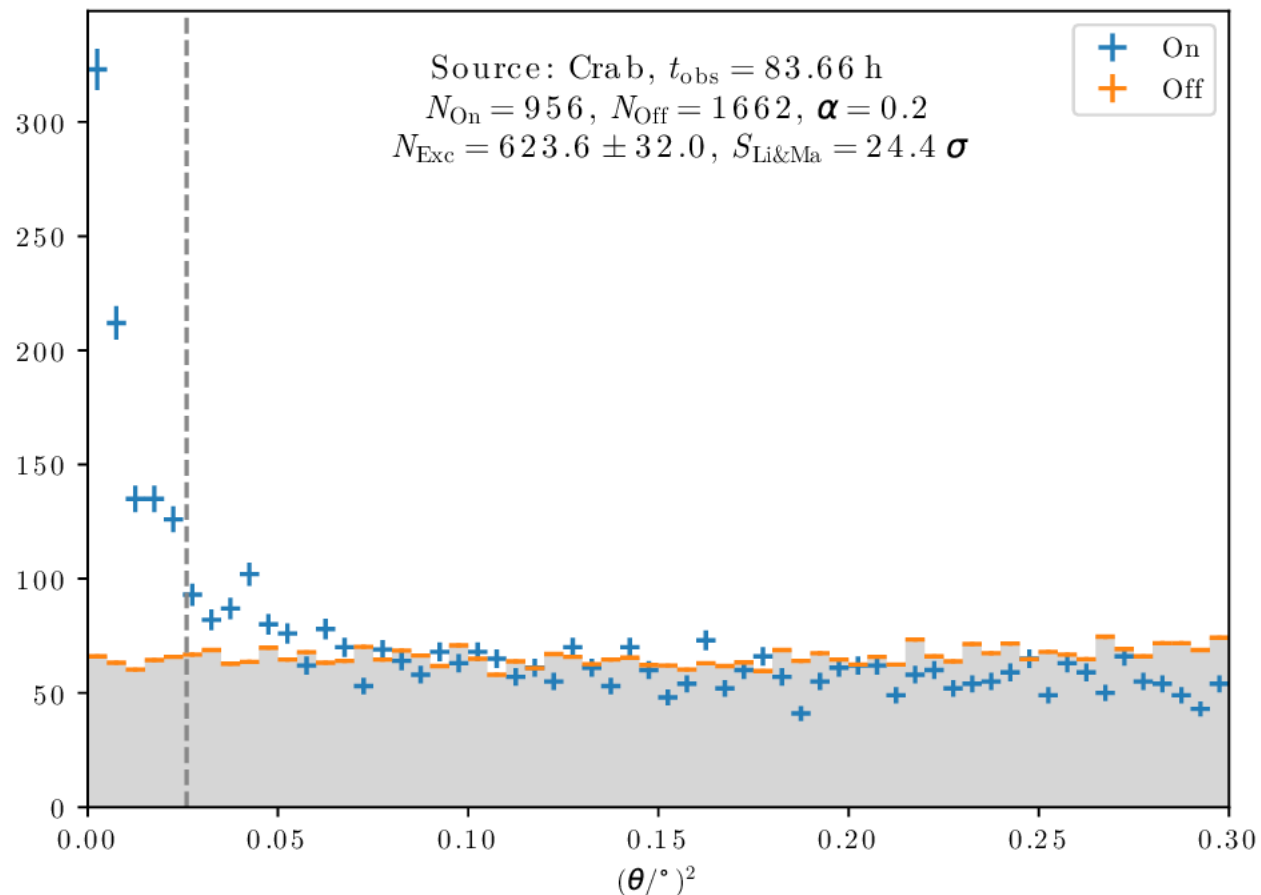
# Eine Gamma/Hadron-Separation ist mit dem Netzwerk möglich

- Monte Carlo: Verhältnis von 1:1 (Hadronen:Gammas)
- Real: Verhältnis 10.000:1 (Hadronen:Gammas)



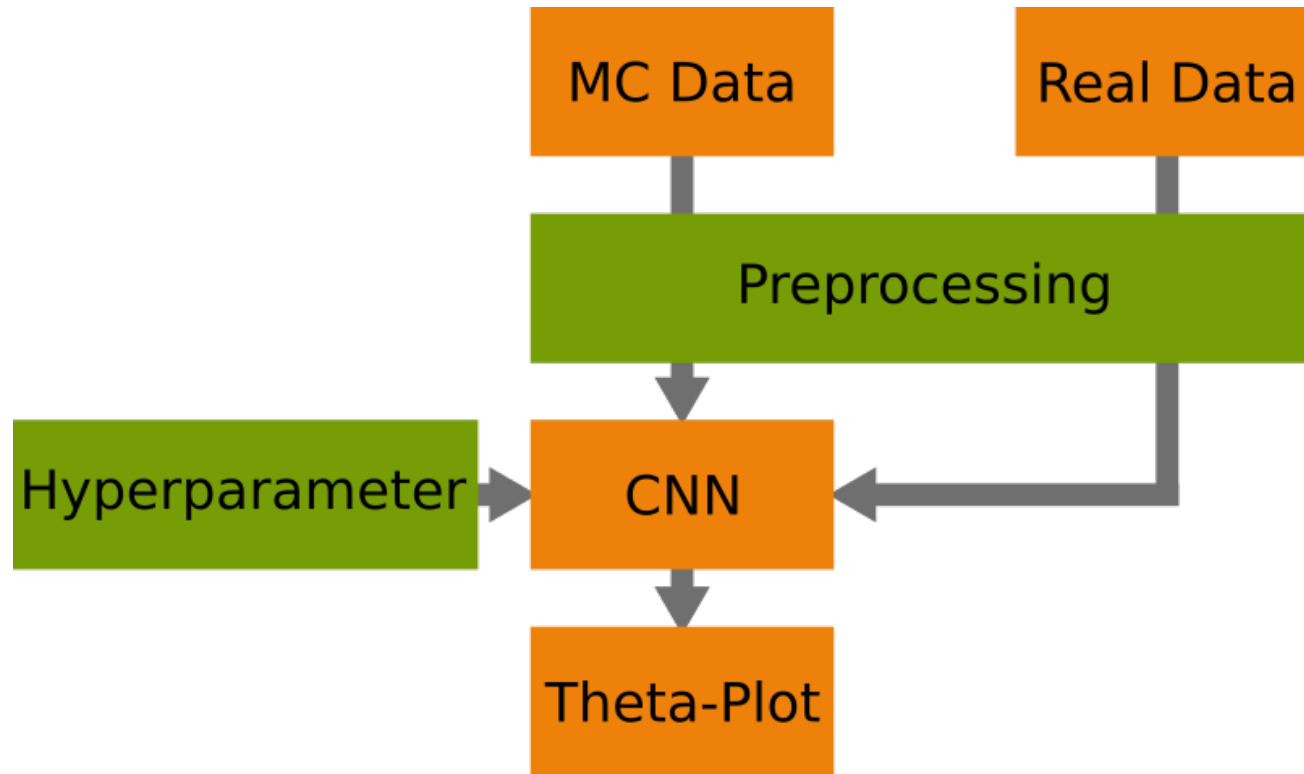
# Der Netzwerk-Signifikanz fehlt ein Faktor 2 für den Krebsnebel

- Zuordnung klassifizierte Photonen zu Himmelsregion
- Vergleich der Aktivität einer Quell-Position ('On') mit 5 'Off'-Positionen
- Verhältnis von 'On'-Gammas zu 'Off'-Gammas bestimmt die Signifikanz der Quelle
- CNN erreicht Signifikanz von 24.4 Sigma (Random Forest 39.89 Sigma).



# Niedrige Signifikanz kann an den Daten und der CNN-Struktur liegen

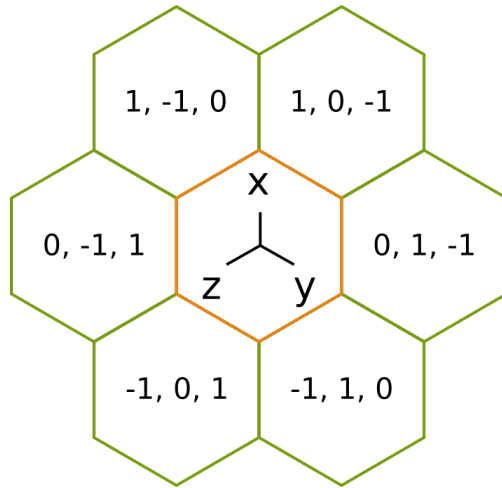
---



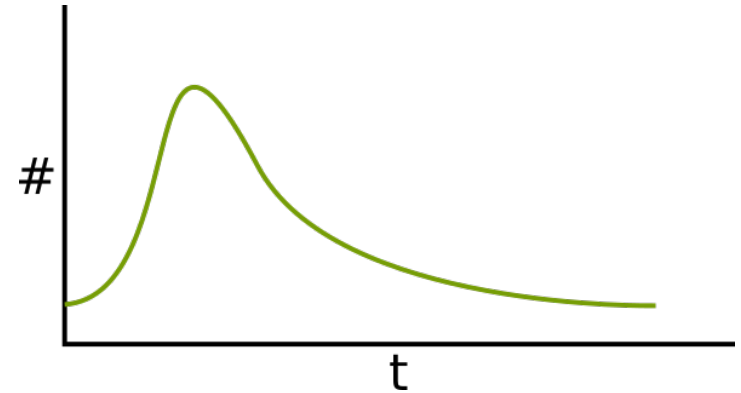


# Die Hexagone und die Zeit könnten bessere Features bieten

- Transformation der hexagonalen Struktur behandelt direkte Nachbarn nicht gleichwertig
- Transformation zu 3D-Tensor behandelt die Nachbarschaften identisch

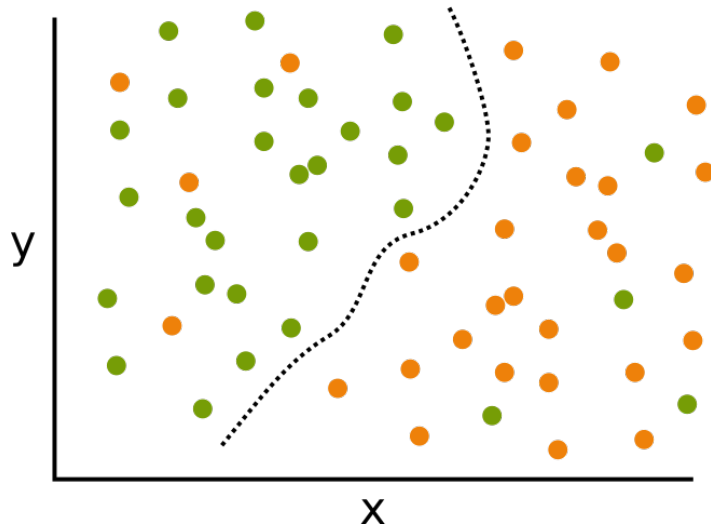


- Aufsummieren der 100 Zeitebenen entfernt die Zeitinformationen
- CNN für 3D-Objekte/Videos kann Zeitinformationen auswerten



# Reale Bilder und evolutionäre Algorithmen

- Simulierte Daten können immer MC-Mismatches enthalten
- Mit Random Forest gelabelte reale Bilder können als Trainingsquelle getestet werden



- Nur wenige Hyperparameter konnten abgesucht werden
- Evolutionäre Algorithmen könnten den Featurespace eigenständig optimieren



# CNNs funktionieren und bieten weiteres Potential

