Particles and Cosmology

16th Baksan School on Astroparticle Physics

# Machine Learning in Astroparticle Physics

Oleg Kalashev
Institute for Nuclear Research, RAS

Lecture 3

**April 10-18, 2019**

# Practical Task
# Classification of gamma-ray sources

**Source:** **FERMI LAT 3FGL catalog**
**Task:** **use source features to predict source class (discriminate between blazars and pulsars)**
**Data:** **~3000 objects ~1000 of which are not identified**

**column description:**

1. **object name (3FGL prefix omitted)**
2. **equatorial coordinates: Right Ascencion, deg**
3. **equatorial coordinates: Declination, deg**

**4-29. spectral and variability parameters**

30. **Source type code or NULL for unidentified sources.**
    **Blazars: bll,BLL,bcu,BCU,fsrq,FSRQ.**
    **Pulsars: psr, PSR.**

# Time to open jupyter notebook

# Gradient Descent Optimization

$$w = w - \eta \cdot \nabla_w C(w)$$

**Problem: gradient strongly oscillates between mini-batches**

**Possible solution - momentum:**

$$v_t = \gamma v_{t-1} + \eta \nabla_w C(w)$$

$$w = w - v_t \qquad \gamma \simeq 0.9$$

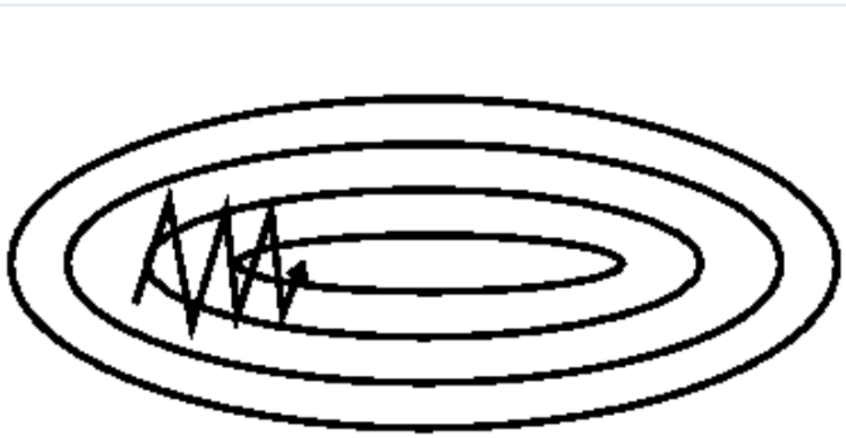


Image 2: SGD without momentum

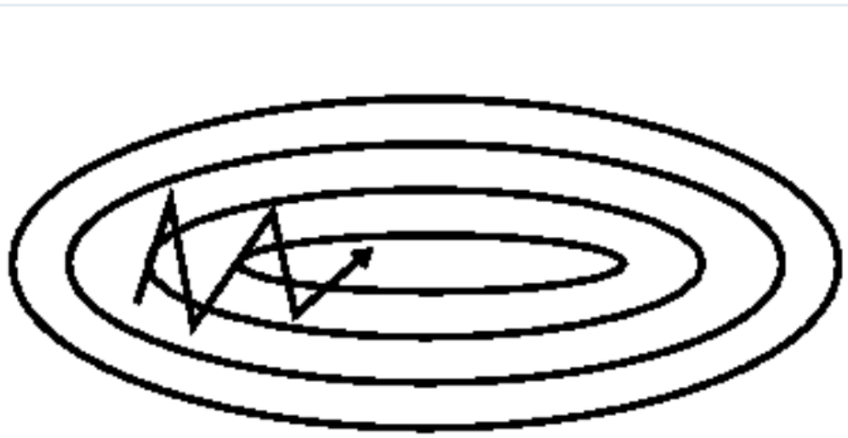


Image 3: SGD with momentum

**Nesterov accelerated gradient (NAG):**
*Nesterov, Y. (1983)*

$$v_t = \gamma v_{t-1} + \eta \nabla_w C(w - \gamma v_{t-1})$$

$$w = w - v_t$$

# Gradient Descent Optimization

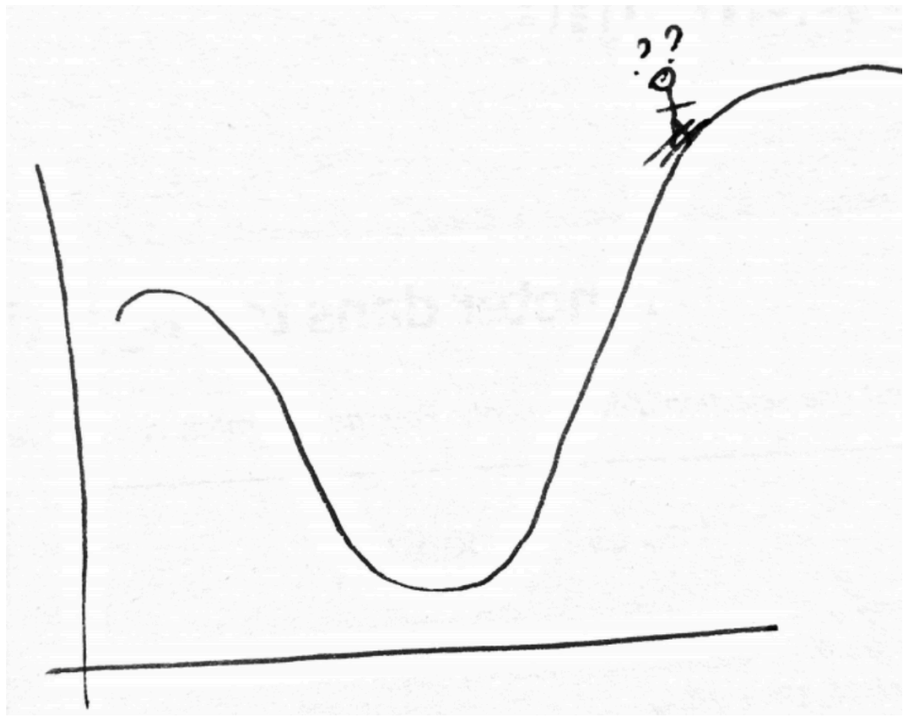
$$w = w - \eta \cdot \nabla_w C(w)$$

**Problem: gradient strongly oscillates between mini-batches**

**Possible solution - momentum:**

$$v_t = \gamma v_{t-1} + \eta \nabla_w C(w)$$

$$w = w - v_t \qquad \gamma \simeq 0.9$$



**helps to skip the local minima**

**exercise:**
**test the method on mt. Elbrus slopes**

**Nesterov accelerated gradient (NAG):**
*Nesterov, Y. (1983)*

$$v_t = \gamma v_{t-1} + \eta \nabla_w C(w - \gamma v_{t-1})$$

$$w = w - v_t$$

# Gradient Descent Optimization

**Idea: adapt our updates to each individual parameter to perform smaller updates for parameters associated with frequently occurring features, and larger updates for parameters associated with infrequent features**

## AdaDelta - Adaptive Learning Rate Method *Zeiler (2012)*

$$\Delta w_t = -\frac{RMS[\Delta w]_{t-1}}{RMS[\nabla_w C]_t} \nabla_w C_t$$

$$w_{t+1} = w_t + \Delta w_t \qquad\qquad RMS[\nabla_w C] \equiv \sqrt{E[(\nabla_w C)^2]}$$

$RMS[\nabla_w C]$ and $RMS[\Delta_w]$ could be estimated either using last *N* points or as a decaying average:

$$E[(\nabla_w C)^2]_t = \gamma E[(\nabla_w C)^2]_{t-1} + (1-\gamma)(\nabla_w C)_t^2 \qquad \gamma \simeq 0.9$$

# Time to open jupyter notebook

# Ways to avoid overfitting (model regularization)

**Overfitting:** **more complex models tend to adjust to particular training examples and lose predictive power**

**exercise:**
**plot the distribution of the weight absolute values in the overfitted model**

**L1, L2 regularisation:** **introduce extra penalty for large weights in loss function**

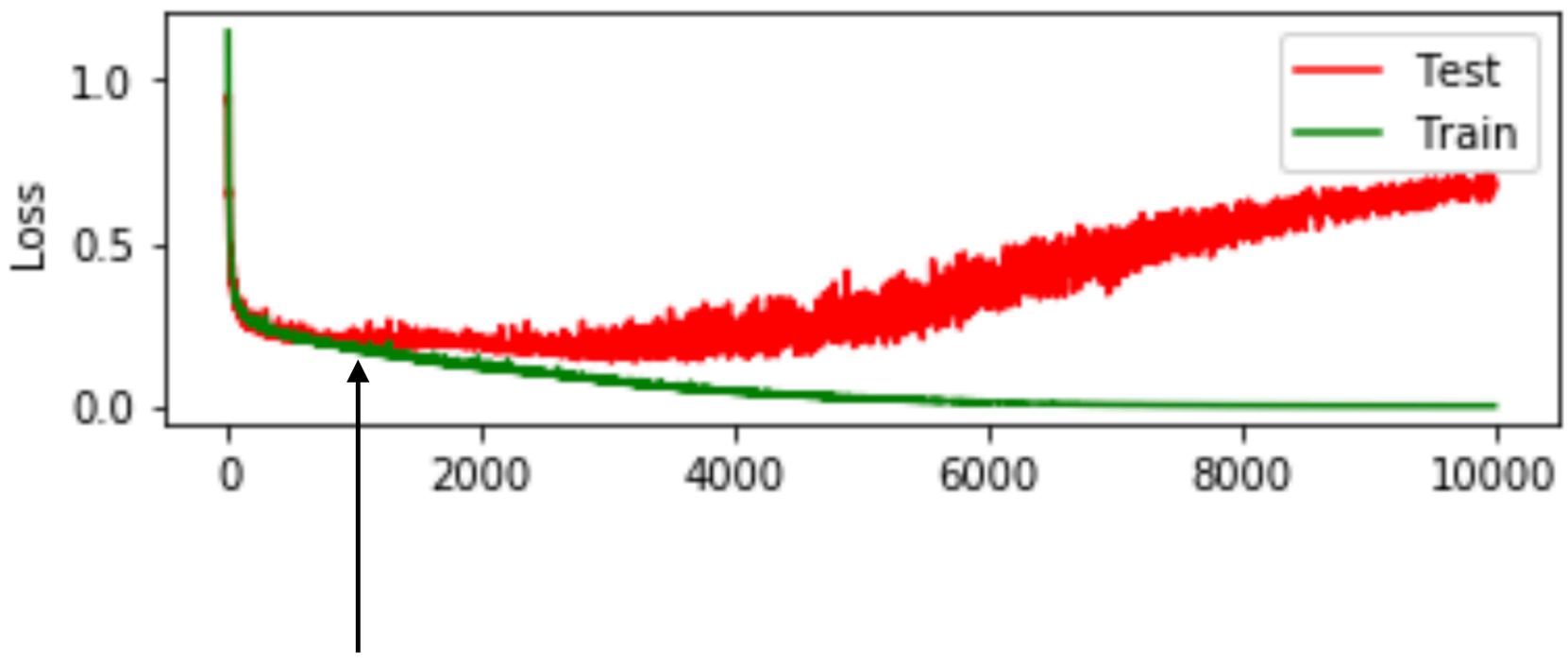
$$C = \ldots + \lambda_1 \frac{1}{N_w} \sum_w |w|$$

$$C = \ldots + \lambda_2 \frac{1}{N_w} \sum_w w^2$$

$N_w$ **- number of weights in the model**

# Ways to avoid overfitting (model regularization)

**Early Stop: monitor loss/accuracy on separate validation data and stop training when it begins to drop**

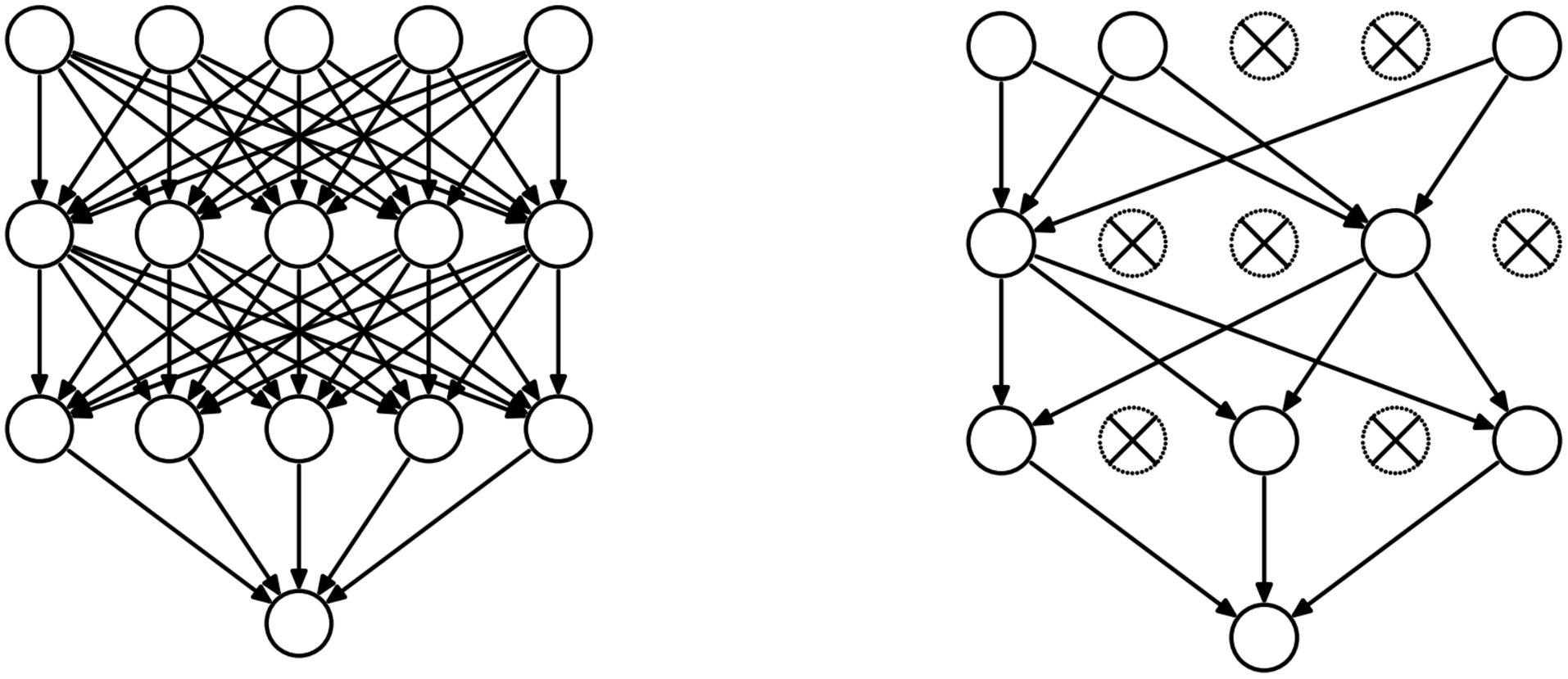



**stop here**

# Regularization

## **Dropout:** randomly disable fraction of neurons when training

Srivastava et. al JMLR 15 (2014)



- In training mode neurons are switched off with probability *p*
- *For p=0.5* we train simultaneously $2^n$ thinned neural networks
- In prediction mode neurons are on but their output weights are multiplied by *p (we average predictions of thinned nets)*

# L1,L2 regularization
# Bayesian interpretation

**Bayes Theorem:** $P(w|y) = \frac{P(y|w)P(w)}{P(y)}$ $\quad\quad$ $\text{posterior} = \frac{\text{likelihood} \cdot \text{prior}}{\text{evidence}}$

**Maximum a posteriori probability estimate (MAP):**

$$\hat{w}_{\text{MAP}} = \arg\max_w P(w|y) = \arg\max_w \frac{P(y|w)P(w)}{P(y)}$$

$$= \arg\max_w P(y|w)P(w) = \arg\max_w \log(P(y|w)P(w))$$

$$= \arg\max_w [\log P(y|w) + \log P(w)]$$

**Regularization can be interpreted as prior** $\log P(w)$

**L2 - regularisation:** $P(w) \propto e^{-\frac{w^2}{2\sigma^2}}$ **(normal distribution)**

**L1 - regularisation:** $P(w) \propto e^{-\frac{w}{b}}$ **(Laplace distribution)**