Deep Learning

Big Data & Machine Learning Bootcamp - Keep Coding

Outline

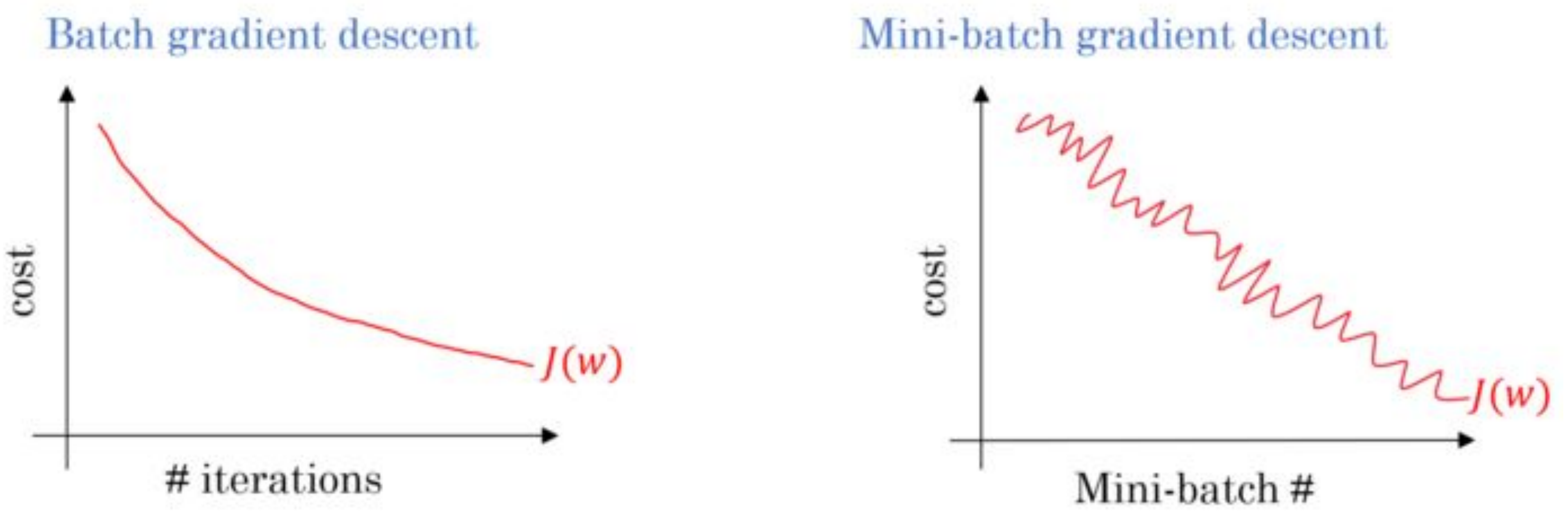
1. Mini batch gradient descent
2. Gradient descent with momentum
3. Adam optimization algorithm
4. Hyperparameters
5. Softmax Regression
6. Deep Learning Frameworks

Mini batch gradient descent

One epoch is a single pass through the training set!

Using batch gradient descent, you only take one step per epoch.

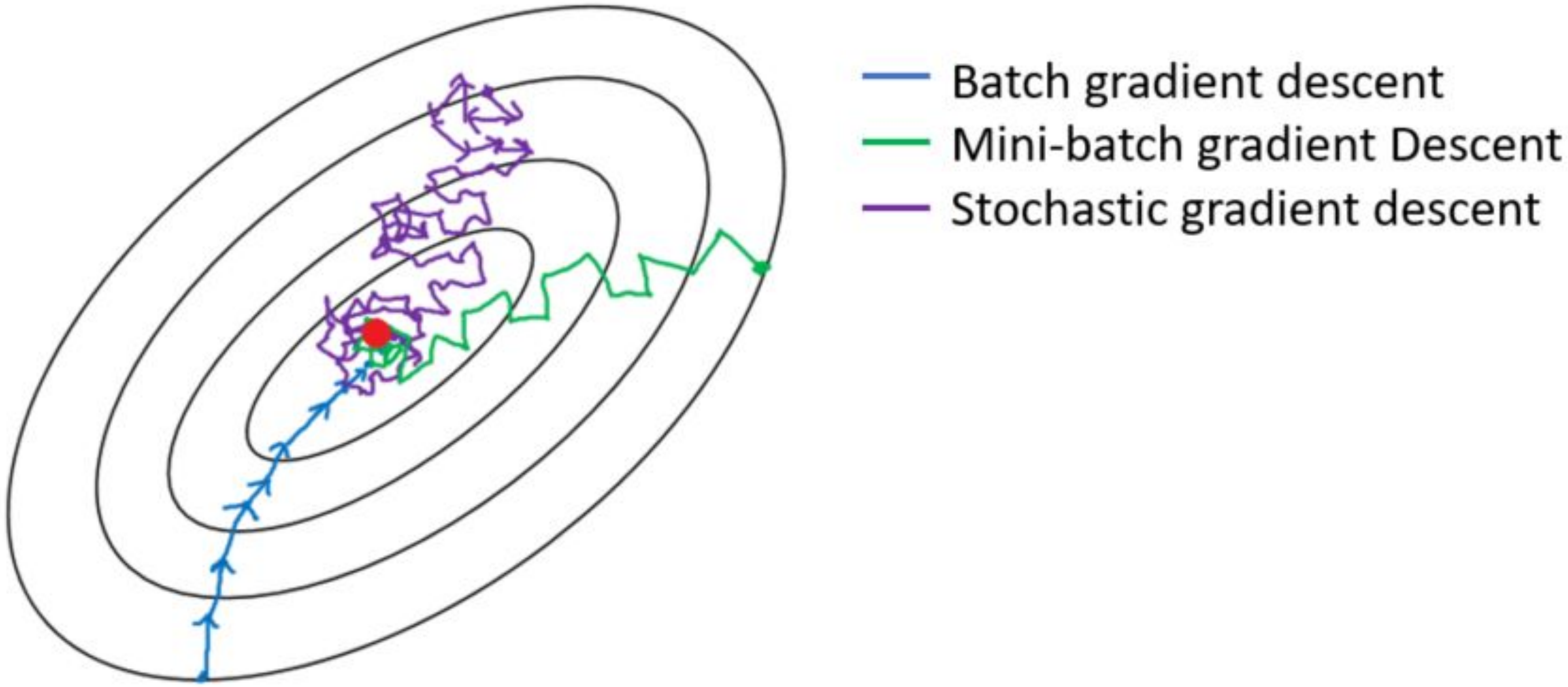
Using mini batch gradient descent, you take #batches steps per epoch



Sources:
- Coursera
- https://towardsdatascience.com/gradient-descent-algorithm-and-its-variants-10f652806a3

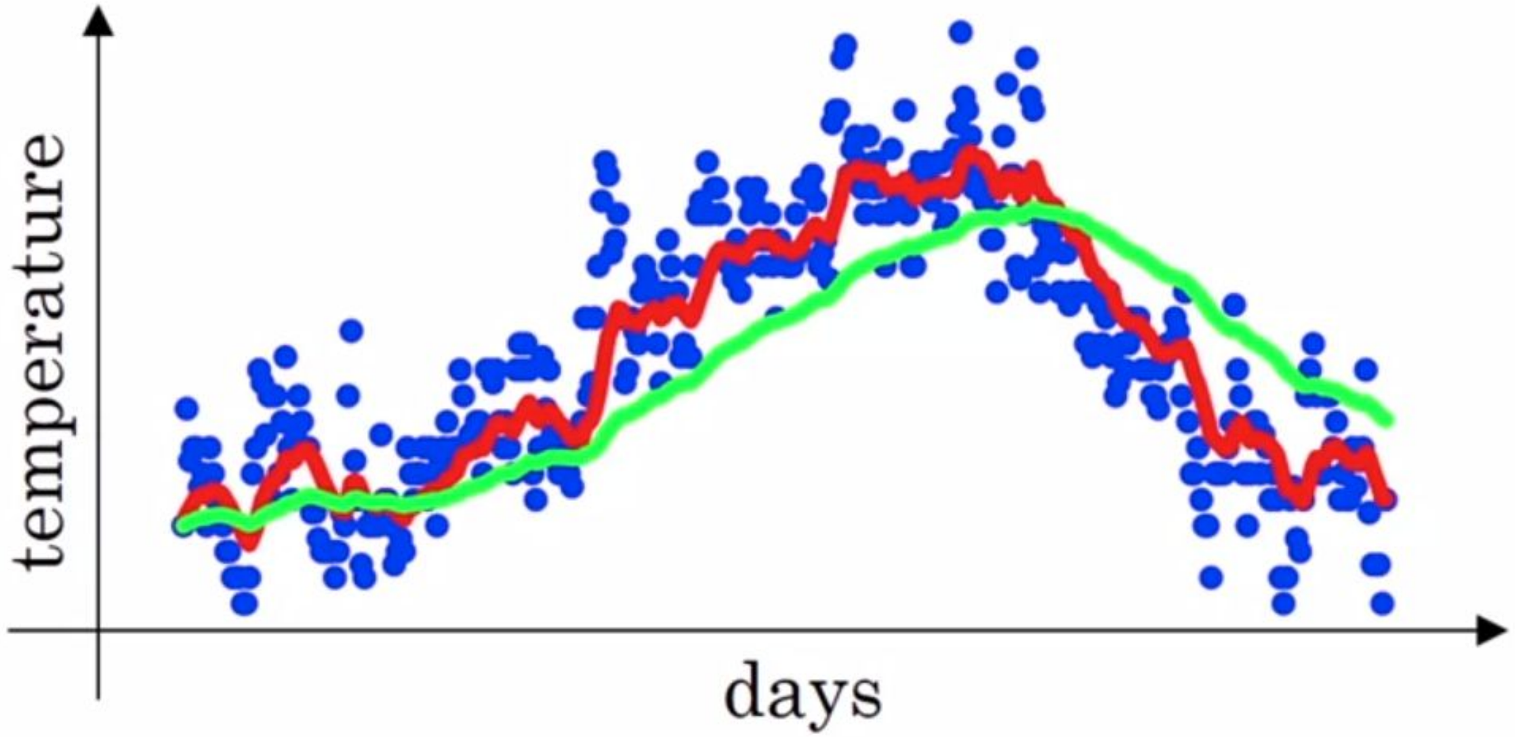
Mini batch gradient descent

Fastest learning occurs when using stochastic gradient descent!



Sources:
- Coursera
- https://towardsdatascience.com/gradient-descent-algorithm-and-its-variants-10f652806a3

Gradient descent with momentum



$$v_t = \beta v_{t-1} + (1 - \beta)\theta_t$$

Blue dots: Data

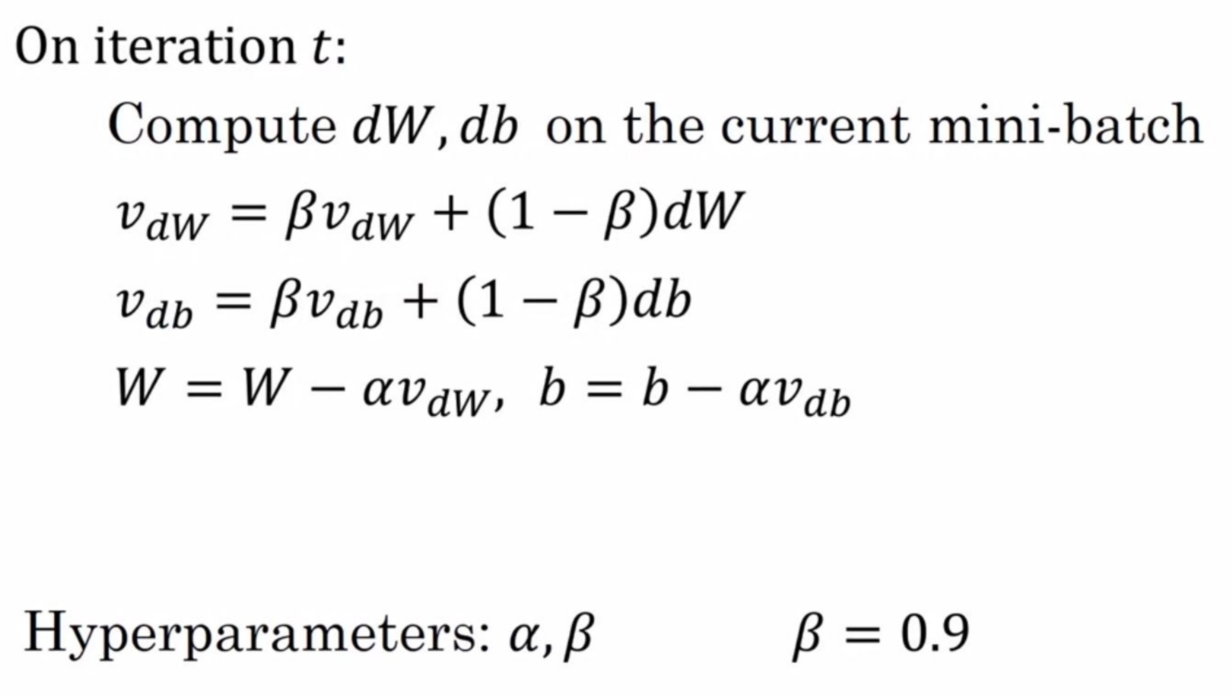
*Red line: $\beta = 0.9$ (**Intuition**: You focused on the last 10 values)*

*Blue line: $\beta = 0.98$ (**Intuition:** You focused on the last 50 values)*

Sources:

- Coursera

Gradient descent with momentum



```
On iteration $t$:
    Compute $dW, db$ on the current mini-batch
    $v_{dW} = \beta v_{dW} + (1-\beta)dW$
    $v_{db} = \beta v_{db} + (1-\beta)db$
    $W = W - \alpha v_{dW}, \; b = b - \alpha v_{db}$
```

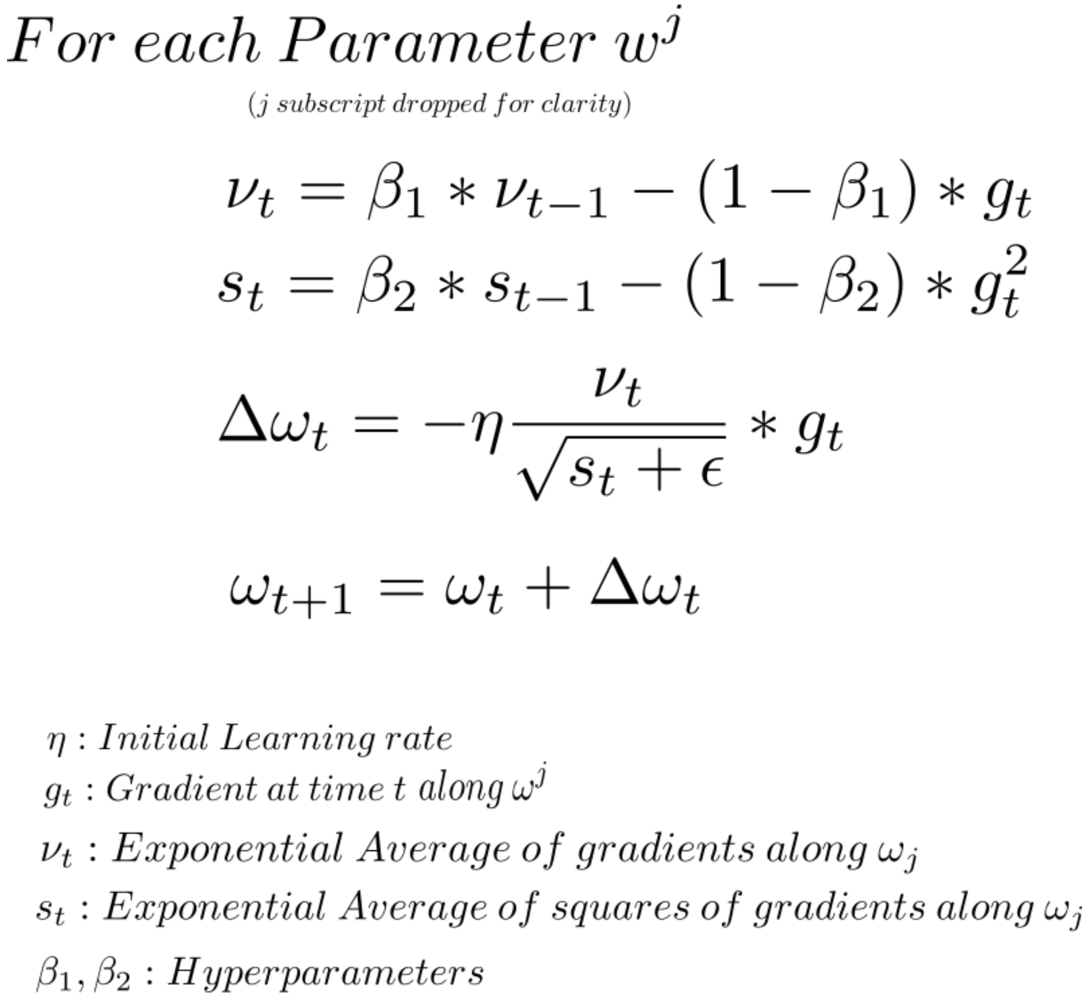
Hyperparameters: α, β $\beta = 0.9$

Sources:
- Coursera

Adam optimization algorithm

Adam or Adaptive Moment Estimation is probably the most used optimization algorithm!

It is basically a **combination of RMSProp and gradient descent with momentum** optimization algorithms!



$$For\ each\ Parameter\ w^j$$

$(j\ subscript\ dropped\ for\ clarity)$

$$\nu_t = \beta_1 * \nu_{t-1} - (1 - \beta_1) * g_t$$

$$s_t = \beta_2 * s_{t-1} - (1 - \beta_2) * g_t^2$$

$$\Delta\omega_t = -\eta \frac{\nu_t}{\sqrt{s_t + \epsilon}} * g_t$$

$$\omega_{t+1} = \omega_t + \Delta\omega_t$$

η : $Initial\ Learning\ rate$

g_t : $Gradient\ at\ time\ t\ along\ \omega^j$

ν_t : $Exponential\ Average\ of\ gradients\ along\ \omega_j$

s_t : $Exponential\ Average\ of\ squares\ of\ gradients\ along\ \omega_j$

β_1, β_2 : $Hyperparameters$

Sorry for the change in the nomenclature here :)

The essence is that there are three hyperparameters when using Adam: learning rate, β_1 and β_2.

β_1 is commonly set to 0.9 and β_2 is commonly set in 0.99

Sources:
- Coursera
- https://blog.paperspace.com/intro-to-optimization-momentum-rmsprop-adam/

Hyperparameters

So far we have talked about many hyperparameters:

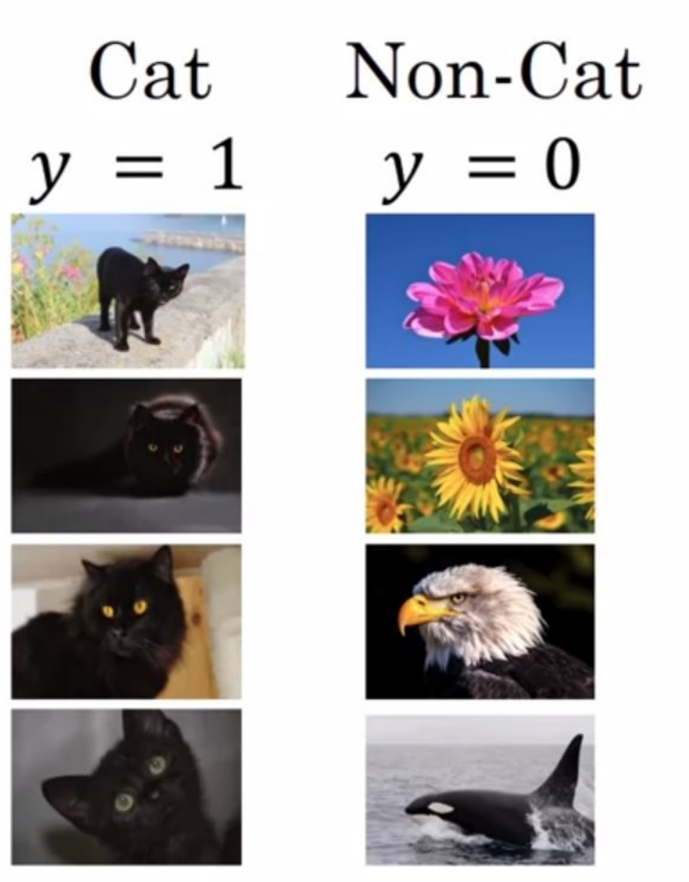
- Learning rate
- Momentum (~0.9)
- β_1, β_2 (Adam optimizer)
- Mini batch size
- Number of layers
- Number of hidden units
- Learning rate decay

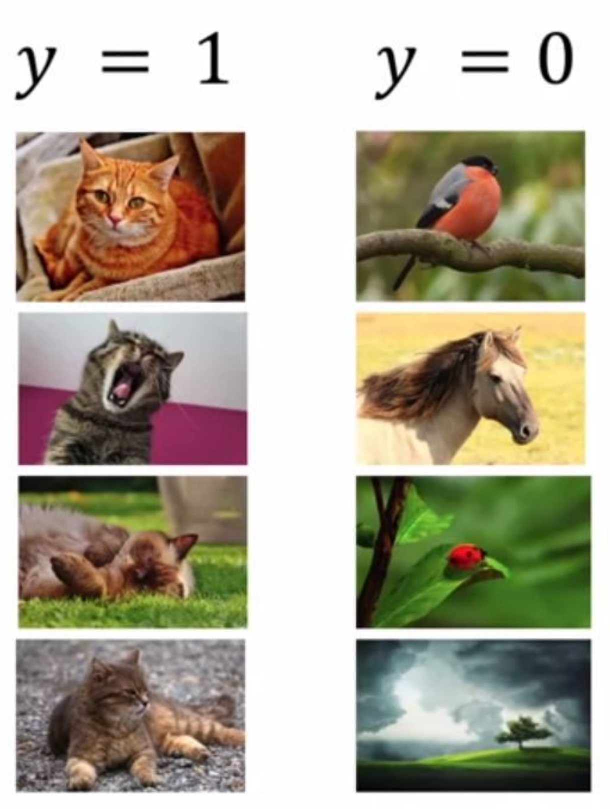
Color means importance of the hyperparameters (Red: very important, Orange: mid importance and Green: low importance)

Sources:
- Coursera

Hyperparameters - Batch Normalization

But why **Batch Normalization (BN)** helps? Basically it makes the neural network robust to **“covariate shift”**
Say you train a neural network on black cats only. If we don’t use BN, the performance of the system will be bad when testing on cats of different colors!

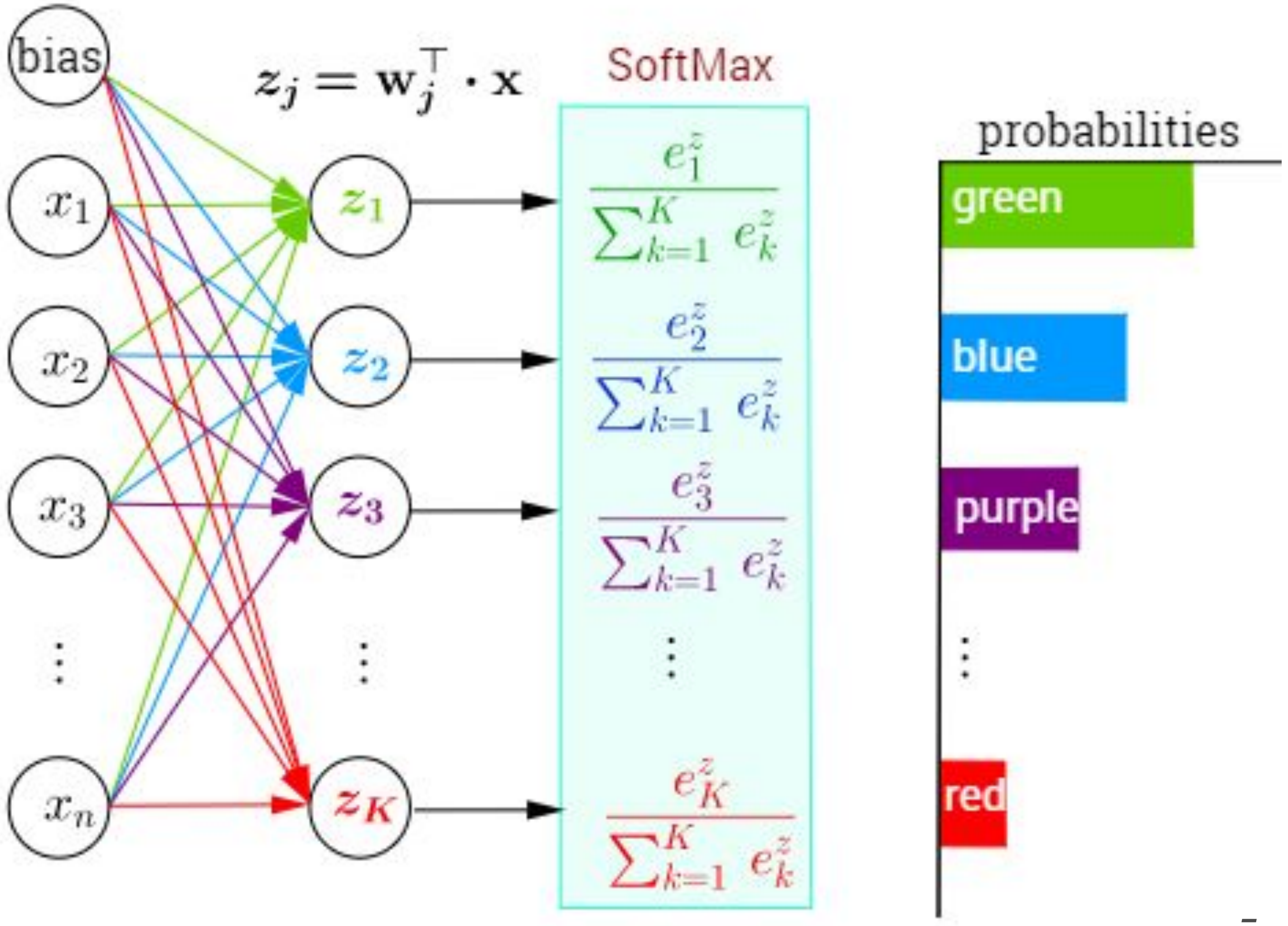


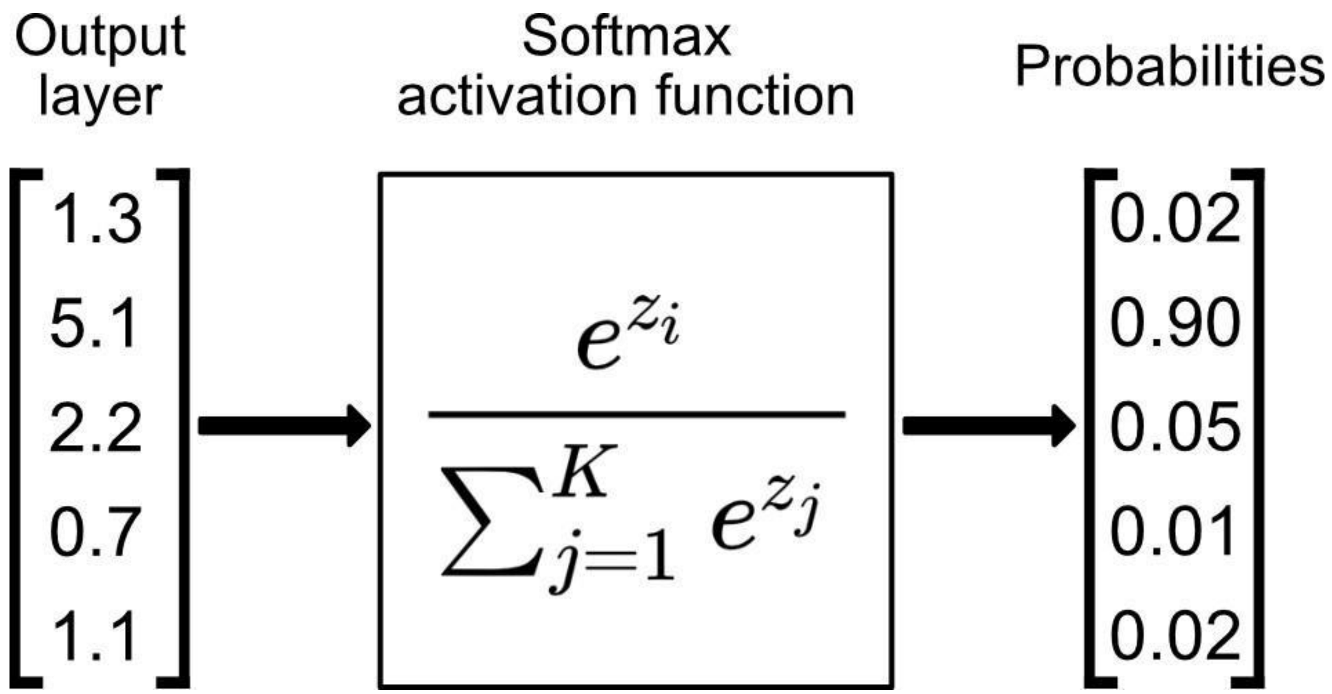


Sources:
- Coursera
- Ioffe, S. and Szegedy, C., 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift.

Softmax Regression

The Softmax layer outputs probabilities:


bias
x_1
x_2
x_3
x_n
$z_j = \mathbf{w}_j^\top \cdot \mathbf{x}$
z_1
z_2
z_3
z_K
SoftMax
$\frac{e^{z}_1}{\sum_{k=1}^{K} e^{z}_k}$
$\frac{e^{z}_2}{\sum_{k=1}^{K} e^{z}_k}$
$\frac{e^{z}_3}{\sum_{k=1}^{K} e^{z}_k}$
$\frac{e^{z}_K}{\sum_{k=1}^{K} e^{z}_k}$
probabilities
green
blue
purple
red


Output layer
1.3
5.1
2.2
0.7
1.1
Softmax activation function
$\frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$
Probabilities
0.02
0.90
0.05
0.01
0.02

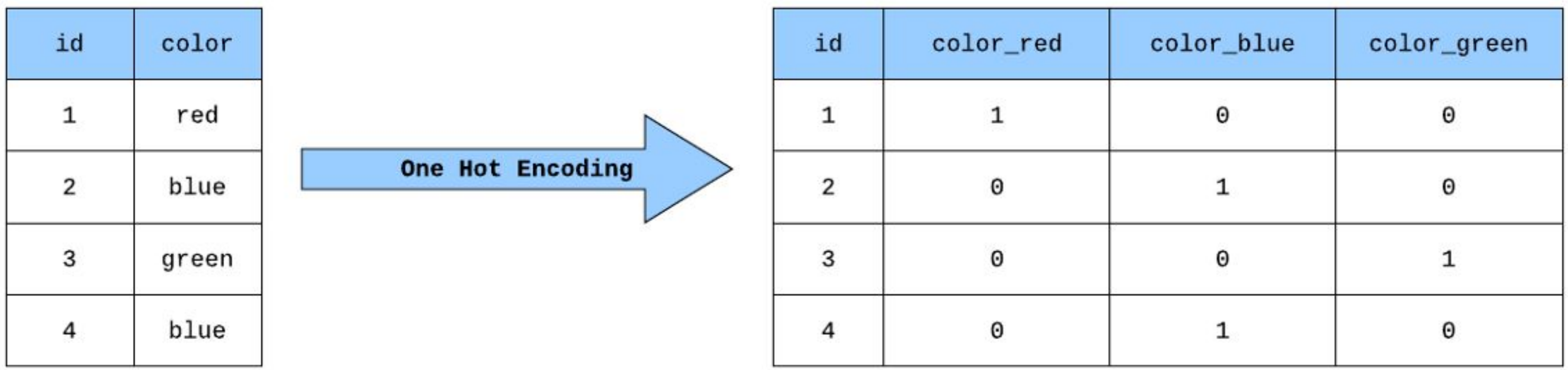
- *K is the number of classes!*
- *The output probabilities sum up to 1 as a probability distro does*

Sources:
- https://www.pinterest.co.uk/pin/849702654687897076/
- https://towardsdatascience.com/softmax-activation-function-explained-a7e1bc3ad60

Softmax Regression

How do you encode the labels/classes when using softmax classification?

One-hot encoding



id	color
1	red
2	blue
3	green
4	blue

One Hot Encoding

id	color_red	color_blue	color_green
1	1	0	0
2	0	1	0
3	0	0	1
4	0	1	0

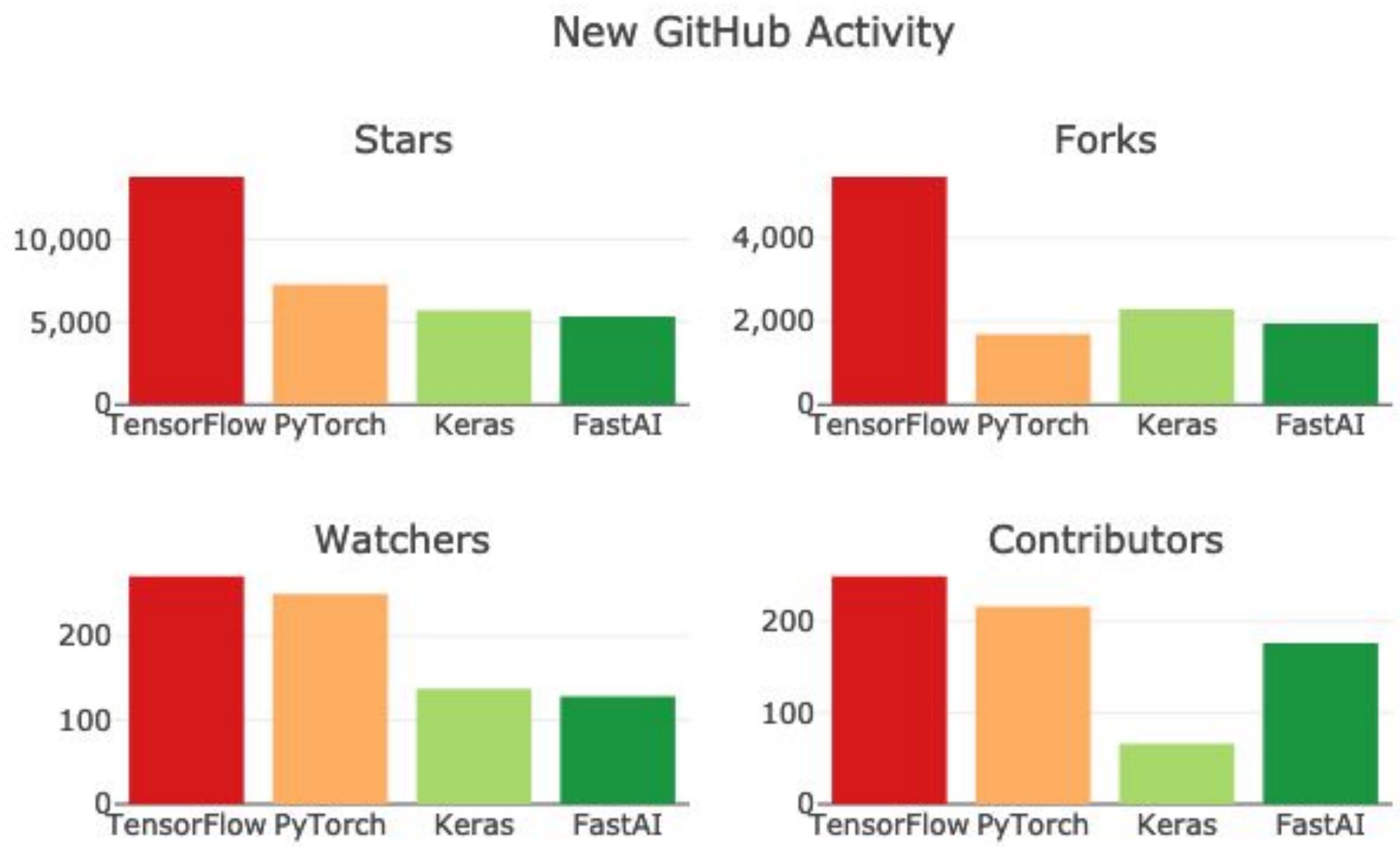
Sources:
- Coursera
- https://towardsdatascience.com/building-a-one-hot-encoding-layer-with-tensorflow-f907d686bf39

Deep Learning Frameworks

The criteria to choose a deep learning framework is:

- Ease of programming (development and deployment)
- Running speed
- Open source

TensorFlow: https://www.tensorflow.org/
PyTorch: https://pytorch.org/
Keras: https://keras.io/
FastAI: https://www.fast.ai/



Sources:
- Coursera
- https://www.kdnuggets.com/2019/05/which-deep-learning-framework-growing-fastest.html