Optimization

- Using gradient descent;
- Develop gradient descent algorithm

Types of gradient descent algorithms

- Batch gradient descent: Using one big training set to train NN
- Mini-batch gradient descent: Using little fragments of training set to train NN (much faster less noisy)
- Stochastic gradient descent: Using one sample of training set at each epoch to train NN (much noisy)
- {i} the ith mini-batch / ith epoch

Tips

- For small training set (m <= 2000) use batch gradient descent

- Mini batch size == 2^n (hyperparameter)

Exponentially weighet moving average

- $V_t = \beta V_{t-1} + (1-\beta)\theta_t$
- For $\beta \to 1$: value estimated from last values
- For $\beta \to 0$: value estimated from current value
- Generally: V_t as approximatif average over $\frac{1}{1-\beta}$ days temperature
- β is an hyperparameter.
- Bias correction: make calculation of EWMA more close to what should be expected at first points of function.
- Bias correction: $V_t \coloneqq \frac{V_t}{1-\beta^t}$

Gradient descent with momentum

- Use Exponentially Weighted Moving Average in calculating of dw and db for each mini-batch

- $V_{dw} = \beta V_{dw} + (1-\beta)dw$ (some literature $V_{dw} = \beta V_{dw} + dw$)
- $V_{db} = \beta V_{db} + (1-\beta)db$

- $w = w - \alpha V_{dw}$
- b $= b - \alpha V_{db}$

- Bias correction: $\frac{V_{dw}}{1-\beta^t}$ and $\frac{V_{db}}{1-\beta^t}$

Root mean square prop (RMS prop)

- Allow us to use high learning rate without compromising stability => learning faster
- $S_{dw} = \beta_2 S_{dw} + (1 - \beta_2) dw^2$ (element-wise) so small for higher change of w
- $S_{db} = \beta_2 S_{db} + (1 - \beta_2) db^2$ so high for smaller change of b

- $w = w - \alpha \frac{dw}{\sqrt{S_{dw} + \epsilon}}$ $(\epsilon = 10^{-8} for\ non\ zeros\ division)$
- $b = b - \alpha \frac{db}{\sqrt{S_{db} + \epsilon}}$

Adam Optimization algorithm

- Adam = RMS + Gradient descent with momentum
- Initialization Vdw = 0; Vdb = 0, Sdw = 0, Sdb = 0
- $V_{dw} = \beta_1 V_{dw} + (1 - \beta_1)dw$
- $V_{db} = \beta_1 V_{db} + (1 - \beta_1)db$
- $S_{dw} = \beta_2 S_{dw} + (1 - \beta_2)dw^2$
- $S_{db} = \beta_2 S_{db} + (1 - \beta_2)db^2$
- Bias correction: $V_{dw}^{corrected} = \frac{V_{dw}}{1-{\beta_1}^t}$ and$V_{db}^{corrected} = \frac{V_{db}}{1-{\beta_1}^t}$
- $S_{dw}^{corrected} = \frac{S_{dw}}{1-{\beta_2}^t}$ and$S_{db}^{corrected} = \frac{S_{db}}{1-{\beta_2}^t}$

Final equations

- $w = w - \alpha \frac{V_{dw}^{corrected}}{\sqrt{S_{dw}^{corrected} + \epsilon}}$
- b= $b - \alpha \frac{V_{db}^{corrected}}{\sqrt{S_{db}^{corrected} + \epsilon}}$

Hyperparameter tuning

- α try to change
- $\beta_1 = 0{,}9$
- β_2 = 0,999
- $\epsilon = 10^{-8}$

- ADAM = ADAptative Moment estimation