Deep Learning Lecture 1 - Compiling Deep Learning Models in Keras

MA8701 General Statistical Methods

Thiago G. Martins, Department of Mathematical Sciences, NTNU

Spring 2019

Compiling Keras models

Now that we have a Keras `model` defined. We need to configure how this model will be trained.

Fully connected model for MNIST

```
model %>% compile(
  optimizer = "rmsprop",
  loss = "categorical_crossentropy",
  metrics = c("accuracy")
)
```

Fully connected model for IMDB

```
model %>% compile(
  optimizer = "rmsprop",
  loss = "binary_crossentropy",
  metrics = c("accuracy")
)
```

Fully connected model for Boston House dataset

```
model %>% compile(
  optimizer = "rmsprop",
  loss = "mse",
  metrics = c("mae")
)
```

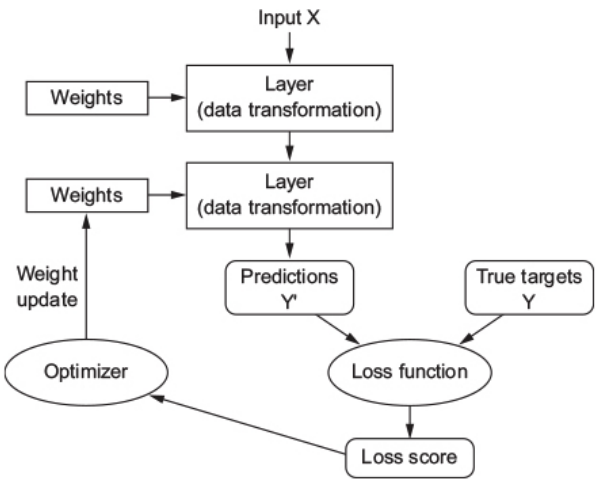
Gradient-based optimization

Gradient descent

- first-order iterative optimization algorithm for finding the minimum of a function `f(x)`.
- `x = x - step * gradient`

Mini-batch stochastic gradient descent (SGD) applied to a Deep Learning model:

1. Draw a batch of training samples `x` and corresponding targets `y`.
2. Run the model on `x` to obtain predictions `y'` (forward pass).
3. Compute the loss of the model on the batch, a measure of the mismatch between `y'` and `y`.
4. Compute the gradient of the loss with regard to the model's parameters (backward pass).
5. `W = W - (step * gradient)`
6. Repeat 1-5 until convergence.



- The algorithm defined above is called mini-batch SGD. The Stochastic part comes from the fact that we are randomly sampling batches `x` from the training data.
- Stochastic gradient descent happens when the batch size equals to 1.
- Mini-batch SGD is a compromise between SGD (one sample per iteration) and full GD (full dataset per iteration)

Backpropagation algorithm:

- DL models take advantage of the fact that all operations used in the model are differentiable.
- Combining the information above with the chain rule of differentiation leads to the Backpropagation algorithm.

- Backpropagation starts with the final loss value and works backward from the top layers to the bottom layers, applying the chain rule to compute the contribution that each parameter had in the loss value.

Variations of SGD

- There exists **many** variations of SGD.
- All our examples use `rmsprop`, which is claimed to be a good default choice.

RMSprop

- Divide the gradient by a running average of its recent magnitude
- `W = W - (step * gradient/rms)`
- `rms = 0.9 * rms + 0.1 * gradient^2`
- Intuition will be give in the lecture.

Further reading:

- Keras documentation for Optimizers
- An overview of gradient descent optimization algorithms
- Overview of mini-batch gradient descent
- Improving Deep Neural Networks: Hyperparameter tuning, Regularization and Optimization

Loss functions

Loss function or objective function:

- The quantity that will be minimized during training.
- It represents a measure of success for the task at hand.

Common problem types and loss functions:

Problem Type	Last-layer activation	Loss function
Binary classification	`sigmoid`	`binary_crossentropy`
Multiclass classification	`softmax`	`categorical_crossentropy`
Regression	None	`mse`

Binary cross-entropy

$$-y_i \log(p_{i1}) - (1 - y_i) \log(1 - p_{i1})$$

Categorical cross-entropy

$$-\sum_{j=1}^{C} y_i \log(p_{ij})$$

Some observations:

- It is not always possible to directly optimize for the metric that measures success on a problem.

- Loss functions, after all, need to be:
  - computable given only a mini-batch of data or ideally given only a single data point.
  - must be differentiable.

Metrics

Provides different forms to measure how well the predictions are compared with the true values.

- `accuracy` : Average of correct classifications
- `mae` : Mean absolute error.

Reference material

This lecture note is based on (Chollet and Allaire 2018) and the following material:

- Keras documentation for Optimizers
- An overview of gradient descent optimization algorithms
- Overview of mini-batch gradient descent
- Improving Deep Neural Networks: Hyperparameter tuning, Regularization and Optimization

References

Chollet, F., and J. Allaire. 2018. *Deep Learning with R*. Manning Publications. https://books.google.no/books?id=xnIRtAEACAAJ.