Deloitte

Convolutional Neural Networks in insurance

Patrik Lipták 01.11.2019

Agenda

- Deep feed-forward neural networks
- Convolutional neural networks
- R session Parkinson disease spiral drawings
- Practical application Online underwriting and Fraud detection in MTPL insurance

Introduction to Deep Neural Networks

Deep Neural Network Architecture



Deep Neural Network Notation

• *m* hidden neurons z₁, ..., z_m with:

$$\begin{aligned} z_i &= f^{(1)} \left(\sum_j W_{i,j}^{(1)} \, x_j + b_i^{(1)} \right), \qquad i = 1, \dots, m \\ z &= f^{(1)} \big(W^{(1)} x + b^{(1)} \big), \qquad z \in \mathbb{R}^m, W^{(1)} \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^n, b^{(1)} \in \mathbb{R}^m \end{aligned}$$

• Linear combination of derived features z:

$$\mathbf{y} = f^{(2)} (\mathbf{W}^{(2)} \mathbf{z} + \mathbf{b}^{(2)}), \qquad \mathbf{y} \in \mathbb{R}^p, \mathbf{W}^{(2)} \in \mathbb{R}^{p \times m}, \mathbf{b}^{(2)} \in \mathbb{R}^p$$
$$\mathbf{y} = f^{(2)} (\mathbf{W}^{(2)} \mathbf{z} + \mathbf{b}^{(2)}) = f^{(2)} (\mathbf{W}^{(2)} (f^{(1)} (\mathbf{W}^{(1)} \mathbf{x} + \mathbf{b}^{(1)})) + \mathbf{b}^{(2)})$$

- $W^{(1)}, W^{(2)}$ are called matrices of weights (with not necessarily positive elements).
- $b^{(1)}, b^{(2)}$ are so called vectors of biases (intercepts).
- $f^{(1)}$ () is called activation function and $f^{(2)}$ output activation function.

Deep Neural Network Activation function

- Converting an input signal of node into an output signal respectively "activation" of a neuron.
- In general non-linear, monotonically increasing and bounded.
- $f^{(2)}$ depends on the nature of the problem classification: softmax, regression: identity...



Deep Neural Network Training

- We usually have a **training set**, which is assumed to consist of examples generated independently from a data generating distribution.
- The goal of optimization is to match the training set as well as possible.
- However, the main goal of machine learning is to perform well on previously unseen data and to minimize so called generalization error or test error. We typically estimate the generalization error using a test set of examples independent of the training set, but generated by the same data generating distribution.

Training of neural nets is composed of two iterative steps:

1. Forward pass: the information of the inputs flows through the model to produce a prediction.

Based on that, we compute a loss which is sometimes called the cost.

2. **Backward pass**: the information of the error flows backwards through the model. Thereby we use the error values to calculate the gradient of the loss with respect to each weight.

In a final step we update the weights (i.e. "move" them in the direction of the steepest descent of the loss).

Deep Neural Network Loss function

- Our objective is to minimize the loss for a neural network by optimizing its parameters weights.
- A common principle used to design loss functions is the *maximum likelihood principle*.
- For the sake of simplicity, let's denote y = f(x). In the case of binary output, the loss function is the negative log-likelihood of the bernoulli distribution, which is commonly just called the cross entropy:

$$L(\mathbf{y}, f(\mathbf{x})) = -\frac{1}{n} \sum_{i=1}^{n} [y_i \log f(\mathbf{x}) + (1 - y_i) \log(1 - f(\mathbf{x}))]$$

• Another choice for linear regression is the mean squared error:

$$L(\mathbf{y}, f(\mathbf{x})) = \frac{1}{n} \sum_{i=1}^{n} (y_i - f(\mathbf{x}))^2$$

- The term cross-entropy is widely used for the negative log-likelihood of a bernoulli or softmax distribution, but that is a misnomer.
 - Any loss consisting of a negative log-likelihood is a cross-entropy between the empirical distribution of the training data and the probability distribution defined by model!
 - For example, the mean squared error is the cross-entropy between the empirical distribution and a Gaussian model.

Deep Neural Network Gradient descent method

- Let f(x) be an arbitrary, differentiable, unrestricted target function, which we want to minimize.
 - We can calculate the gradient $\nabla f(x)$, which always points in the direction of the **steepest ascent**.
 - Thus $-\nabla f(x)$ points in the direction of the **steepest descent**!
- Standing at a point x_k during minimization, we can improve this point by doing the following step:

$$f(x_{k+1}) = f(x_k) - \alpha \,\nabla f(x_k)$$

"Walking down the hill, towards the valley."

α determines the length of the step and is called step size or in terms of neural networks learning rate.
 To find the optimal *α* we need to look at minimal value.

Deep Neural Network Gradient descent method



10

Deep neural network Backpropagation algorithm example – binary classification

$$L(y, y_{out}) = -(y \log y_{out} + (1 - y) \log(1 - y_{out}))$$

$$f^{(2)} \rightarrow \text{sigmoid}$$

$$y_{out} = \frac{1}{1 + \exp(-W^{(2)}z)}$$

$$W_{11}^{(2)}$$

$$W_{21}^{(2)}$$

$$W_{21}^{(2)}$$

$$W_{21}^{(2)}$$

$$Z_{2, out}$$

$$z_{0ut} = MAX(\mathbf{0}, W^{(1)}x)$$

$$W_{11}^{(1)}$$

$$W_{22}^{(1)}$$

$$W_{21}^{(1)}$$

$$W_{22}^{(1)}$$

$$W_{21}^{(1)}$$

$$W_{22}^{(1)}$$

$$W_{22}^{(1)}$$

$$f(\mathbf{x}) = f^{(2)} \left(W^{(2)} \left(f^{(1)} (W^{(1)} \mathbf{x}) \right) \right)$$

$$W_{11}^{(2)} : \frac{\partial L(y, f(\mathbf{x}))}{\partial W_{11}^{(2)}} = \frac{\partial L(y, f(\mathbf{x}))}{\partial y_{out}} \frac{\partial y_{out}}{\partial y_{in}} \frac{\partial y_{in}}{\partial W_{11}^{(2)}}$$
Updating parameter: $W_{11}^{(2)} \leftarrow W_{11}^{(2)} - \alpha \times \frac{\partial L(y, f(\mathbf{x}))}{\partial W_{11}^{(2)}}$

$$W_{11}^{(1)} : \frac{\partial L(y, f(\mathbf{x}))}{\partial W_{11}^{(1)}} = \frac{\partial L(y, f(\mathbf{x}))}{\partial y_{out}} \frac{\partial y_{out}}{\partial y_{in}} \frac{\partial y_{in}}{\partial z_{1,out}} \frac{\partial z_{1,out}}{\partial z_{1,in}} \frac{\partial z_{1,in}}{\partial W_{11}^{(1)}}$$
Updating parameter: $W_{11}^{(1)} \leftarrow W_{11}^{(1)} - \alpha \times \frac{\partial L(y, f(\mathbf{x}))}{\partial W_{11}^{(1)}}$

*Biases are usually initialized to a constant value, usually 0. Weights are usually initialized to small random values, either with uniform or normal distribution.

Overfitting Regularization

Regularization is any change in the machine learning algorithm that is designed to reduce generalization error but not necessarily its training error.

Regularization is usually needed only if training error and generalization error are different.

Methods:

- Early stopping
- L2, L1 regularization
- Dataset augmentation
- Ensembling
- Dropout
- ...



Deep Neural Network Optimization problem

- Convex optimization problems can be reduced to "finding a local minimum".
- In neural networks we have to deal with non-convex problems or flat regions such as saddle points:



Deep Neural Network Stochastic gradient descent

However, serious consequences can be easily avoided using a technique called gradient clipping.

- The gradient does not specify the optimal step size, but only the optimal direction within an infinitesimal region.
- Gradient clipping simply caps the step size to be small enough that it is less likely to go outside the region where the gradient indicates the direction of steepest descent.
- SGD and its modifications are the most used optimization algorithms for machine learning in general and for deep learning in particular.

Algorithm 1 SGD parameter update at training iteration k	In practice, a common strategy is to decay the learning rate α_{ν} linearly over time	
 require learning rate α_k require initial parameter θ while stopping criterion not met do Sample a minibatch of <i>m</i> examples from the training set {x⁽¹⁾,,x^(m)} 	To choose α : monitor learning curves that plot the objective function as a function of time.	
5: Compute gradient estimate: $\hat{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_{i} L(y^{(i)}, f(x^{(i)}, \theta))$ 6: Apply update: $\theta \leftarrow \theta - \alpha \hat{g}$ 7: end while		

• Other techniques: momentum, algorithms with adaptive learning rates (Adagrad, RMSProp, Adam...)

Practical use Use in insurance

- In principle, neural networks bring massive improvements in structuring of unstructured data.
- On the other side, "traditional" structured insurance data is still not "big" enough, that NN's would outperform complex tree of regression models in predicting on structured data.
- Possible usage:
 - Pricing of non-life insurance (Noll, Salzmann and Wüthrich 2018; Wüthrich and Buser 2018),
 - Analysis of telematics data (Gao, Meng and Wüthrich 2018; Gao and Wüthrich 2017),
 - IBNR Reserving (Kuo 2018),
 - Automation,
 - Classification problems,
 - Mortality forecasting (Hainaut 2018).

Quiz time!

Which of these elements must be specified at the beginning of the backpropagation algorithm?

a) Activation functions

b) Number of hidden layers

c) Weights

d) Loss function

Convolutional Neural Networks

Motivation "New" insurance data



Hart Job 1 Ale - have	- Approx.og. clin.	
Togit-adjust-one	Trate area stress and	A REPARTING A
	Wills Yes, 12 socie 8.60	L-fact lineauer,
Contraction of the local division of the	12. CHECOMMENSION	whereas a
From I support have to draw themes	A NOTE ON THE ADDRESS OF THE OWNER	The Property Supported by Annual States
and StifterL,	A -main parter to come - B	- SOMALT
Contraction of the lot	City Contractor (Line) 1 (2	GRAMA
tomani E-3353 a apatrici, Ph.	and and printer.	tor on 1 Sector Astronomic
BULERI 601	C pierre instantes 10	601 681 694
11+0-0	1 wat found in the	Thata
Man International International	Dr. Court billions of	and an and a second sec
REARING CLAT	18 Ten Mid, - p. Maril als Stor	ZALJ SHAR SAN
25 13 900	13 Pageto to coulor in 2	St. Sines
descriptions /	11 mituration	the participation of the parti
Their Control Control Control of		I holy resource an entry reason
von Bradie Stantanta dest. He	on lange have to cause and	3448 Ryason in 34
COLUMN CODEWAY BA	and the second second	Throad 000 45332140, 05
Animate of pleased and pleased	CALCE B. Charter	Commit d'acces
stream and the second s	D+ marinh HO	
type:	119 and FD	Test of the second seco
Address -	Da -man 10	- theme
W Arrise	S	Mound.
do delle tableci in advanta esta i monte in	Dis anticipation and	The state in case of the state of the state of the
The second s	milet a proper	
- X694T	Da attant att	Man Dama M.
Hattys.	Barris and a	And Address of the second
11.19.192	C Antipartition 2	ter + saper . 35.05,
Anna Tanad Danara	Deman	- Diam paren
iden panal	incastra Para coli.	to an idea present
0013404		Sector and a sector and a sector and a sector a
Anna child passes (13.8% 2000	The state of the state of the second state of	Ta. ci. 2015
Aller a part in a second in the	The second se	Anna Die Judges in sale de des
and the second second second		the second secon
1 NO 1 -1		- 2 -
·日日 →	D.	
24 J	-1300	
	1000	112
14(117)	Tret ER	Be-P-7
"statistication of the lot of the lot	1.34	A Re O'MA INTELL
transfer and the set of the	17.0	USA Built In lan.
T MARKE BARDIC	1.1.4	- Carla Monthe da.
A	Specie de company	Di Calandari Angela
DO STREET P. Br. Mark.	Sul CI.	"It is the Markens' yes of the
te alle liebles, califiture	194 Chail	COLOR AL COLOR





and the second
and a pin
A3-28/13
ALART OL LAS BORD COMMENT ALLOW FOR PORT
RD Nachdagene Brose
The LEGIE, MARADONE & CONTTAL 10 1710
REPORT OF PETERFORMENT COUNTY MEDICAL EXAMPLE
man Streetsthe Dourse also and mark in the
when when when a set of the set o
The Stack Street and and
Will work Caused Land L hand L manual man a cause sain B Road and D
In Profest C. Dupping control of constant (J. Constant 2)
Man device manual manufacture and an entry of the second and the s
Ainespert RIL
Transfer Could Design Free Could
Real and a second
and the law of many CP mainten
the second se
Fam Tegit Te C Land C Line Cole
and a second of
Contract of the second se
VES AVA
Multiple was mounder 71° Nº 12 A
(A. A. (8B)
/// the it
internet and a second s
and the second s
and such as providends \\/\\\
have at bland at server I hauf the
interfer del mun in 11 untrenter 18
alde dist marting
Kendre beremete and bernet 11 128
HORNEY OND D 20-DF HONDE D CRUIK BOOK D CRUIK BOOK D CRUIK
Construction of the second sec
story
The second
and a second second second second
built interaction that are reasoned as some of the short has that have a first distribution of the bady and made something
and the internet of Statistics are been put former to det Order Managettawa the Antonest and the re-
instative particular general specification in the test and merest in the last of we handlings and index.
1 The Color Communitional Wheel I a
and the second white the







Convolutional neural network Concept

Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

How to make it work? Ideas?







Convolved Feature

Convolutional neural network Convolution

Extracting the high-level features/patterns such as edges, colour, gradient orientation from the input image:



Convolutional neural network Pooling

- Decrease the computational power required to process the data.
- Extracting dominant features which are rotational and positional invariant.

Two types of Pooling: Max Pooling (noise suppressant) and Average Pooling:



Convolutional neural network Practical example



22

Quiz time!

Which stage of CNN extract the high-level patterns such as edges... from the input image?

a) Pooling

b) Convolution

R session Parkinson disease spiral drawings

Coding R as a programming language



- Tensorflow
- Keras (include installing 'reticulate' package for interface of Python in R)

List of different types of neural network models that can be built in R using Keras:

- Feed-forward neural networks
- Convolutional neural networks
- Recurrent neural networks
- Use pre-trained models like VGG16, RESNET etc.
- Fine-tune the pre-trained models.

*'devtools' package for installing packages from github needed

R session Syntax of keras package – binary classification

```
model <- keras model sequential() %>%
 layer conv 2d(filters = 8,
               kernel_size = c(5,5),
               activation = 'relu',
               input shape = input shape) \% > \%
 layer_max_pooling_2d(pool_size = c(3, 3)) %>%
 layer dropout(rate = 0.25) %>%
 ### next possible convolutional and pooling layers
 layer_flatten() %>%
 layer_dense(units = 16, activation = 'relu') %>%
 layer dropout(rate = 0.25) %>%
 layer dense(units = 2, activation = 'sigmoid')
model %>% compile(
 loss = "binary crossentropy",
 optimizer = "adam",
 metrics = c('accuracy')
batch size <- 72
epochs <- 50
# Train model
model %>% fit(
 x train, y train,
 batch size = batch size,
 epochs = epochs,
 validation_split = 0.2,
 shuffle = TRUE
# Predictions
model %>% evaluate(x_train, y_train)
model %>% evaluate(x_test, y_test)
```

Practical application Online underwriting in MTPL insurance

Practical application Fraud detection

Convolutional neural network Insurance analytics – fraud detection

Costly **claim processing** with large involvement of personal stuff. **Frauds** amounting on average to 10% of all claim expenditure every year (€4bn in DE).

Searpent (with Deloitte's cooperation):

- Visual car accidents and flooded flats fraud detection system for insurance market.
- Identifies repeated claims and modified images across the market.





Going further...



Claims classification (identification of straight-through processing cases vs requiring special treatment) and improved fraud detection using client's structured data.



Improving claims classification

(identification of straight-through processing cases vs requiring special treatment) and **fraud**

detection using client's unstructured textual data processed by NLP.



Monitoring and producing a **fraud risk score** for **voice interactions** based on speech, behavioral and emotional tendencies

detected during **customer service calls** by **voice analytics**.



Providing and setting up simple and scalable all-in-one **cloud environment** that will enable clients to **integrate all relevant internal and external**

data sources and utilize them for making claims processing more effective and efficient.

Conclusion and discussion

- Deep learning can <u>enhance the predictive power</u> of models built by actuaries.
- Very useful for <u>high-frequency</u> and <u>high-dimensional</u> data.
- Application of deep learning techniques to actuarial problems seems to be <u>rapidly emerging field</u> within actuarial science => appears reasonable to predict more advances in the near-term.
- Deep learning is not a panacea for all modelling issues applied to the wrong domain, deep learning will
 not produce better or more useful results than other techniques.
- <u>Winter might be coming</u> if actuaries do not take the lead in applying deep learning, someone else will.

Thank you for your attention! Contact



Patrik Lipták

Senior Consultant | Actuarial & Insurance Solutions

Deloitte Advisory s.r.o.

Churchill I, Italská 2581/67, 120 00, Prague 2 – Vinohrady, Czech Republic

Reception: +420 246 042 500 | M: +420 778 820 777

pliptak@deloitteCE.com | www.deloitte.cz

Deloitte.

- Gao, G., S. Meng and M. Wüthrich. 2018. Claims Frequency Modeling Using Telematics Car Driving Data. SSRN. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3102371. Accessed: 14 October 2019.
- Gao, G. and M. Wüthrich. 2017. Feature Extraction from Telematics Car Driving Heatmaps. SSRN. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3070069. Accessed: 14 October 2019.
- Goodfellow, I., Bengio Y. and A. Courville. 2016. Deep Learning. http://www.deeplearningbook.Org. Accessed: 14 October 2019
- Hainaut, D. 2018. "A neural-network analyzer for mortality forecast", Astin Bulletin 48(2): 481-508.
- Kuo, K. 2018. "DeepTriangle: A Deep Learning Approach to Loss Reserving"
- Noll, A., R. Salzmann and M. Wüthrich. 2018. Case Study: French Motor Third-Party Liability Claims. SSRN. https://ssrn.com/abstract=3164764 Accessed: 14 October 2019.
- Wüthrich, M. and C. Buser. 2018. Data analytics for non-life insurance pricing. Swiss Finance Institute Research Paper. https://ssrn.com/abstract=2870308. Accessed: 14 October 2019.
- EAA seminar: Mitevski, M. and Thomas J. Neural Networks and Deep Learning in Insurance Theory and Practice

Deloitte.

Deloitte refers to one or more of Deloitte Touche Tohmatsu Limited ("DTTL"), its global network of member firms, and their related entities. DTTL (also referred to as "Deloitte Global") and each of its member firms are legally separate and independent entities. DTTL does not provide services to clients. Please see www.deloitte.com/about to learn more.

Deloitte is a leading global provider of audit and assurance, consulting, financial advisory, risk advisory, tax and related services. Our network of member firms in more than 150 countries and territories serves four out of five Fortune Global 500® companies. Learn how Deloitte's approximately 264,000 people make an impact that matters at www.deloitte.com.

This communication contains general information only, and none of Deloitte Touche Tohmatsu Limited, its member firms, or their related entities (collectively, the "Deloitte Network") is, by means of this communication, rendering professional advice or services. Before making any decision or taking any action that may affect your finances or your business, you should consult a qualified professional advisor. No entity in the Deloitte Network shall be responsible for any loss whatsoever sustained by any person who relies on this communication.

© 2019. For information, contact Deloitte Czech Republic.