Appendix A

Examples

- Downloading Examples
- Structure of Example Download
- Keeping Updated

A.1 Artificial Intelligence for Humans

These examples are part of a series of books that is currently under development. Check the website to see which volumes have been completed and are available:

http://www.heatonresearch.com/aifh

The following volumes are planned for this series:

- Volume 0: Introduction to the Math of AI
- Volume 1: Fundamental Algorithms
- Volume 2: Nature-Inspired Algorithms
- Volume 3: Deep Learning and Neural Networks

A.2 Latest Versions

In this appendix, we describe how to obtain the *Artificial Intelligence for Humans* (AIFH) book series examples.

This area is probably the most dynamic of the book. Computer languages are always changing and adding new versions. We will update the examples as it becomes necessary, fixing bugs and making corrections. As a result, make sure that you are always using the latest version of the book examples.

Because this area is so dynamic, this file may become outdated. You can always find the latest version at the following location:

https://github.com/jeffheaton/aifh

A.3 Obtaining the Examples

We provide the book's examples in many programming languages. Core example packs exist for Java, C#, C/C++, Python, and R for most volumes. Volume 3, as of publication, includes Java, C#, and Python. Other languages, such as R and C/C++ are planned. We may have added other languages since publication. The community may have added other languages as well. You can find all examples at the GitHub repository:

https://github.com/jeffheaton/aifh

You have your choice of two different ways to download the examples.

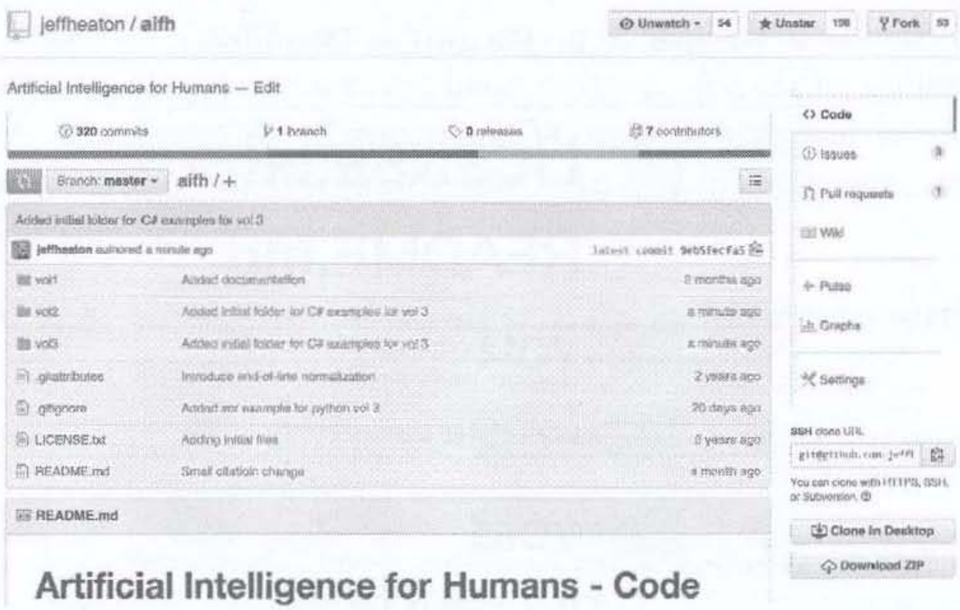
A.3.1 Download ZIP File

GitHub provides an icon that allows you to download a ZIP file that contains all of the example code for the series. A single ZIP file has all of the examples for the series. As a result, we frequently update the contents of this ZIP. If you are starting a new volume, it is important that you verify that you have the latest copy. You can perform the download from the following URL:

https://github.com/jeffheaton/aifh

You can see the download link in Figure A.1:

Figure A.1: GitHub



A.3.2 Clone the Git Repository

You can obtain all the examples with the source control program **git** if it is installed on your system. The following command clones the examples to your computer: (Cloning simply refers to the process of copying the example files.)

```
git clone https://github.com/jeffheaton/aifh.git
```

You can also pull the latest updates with the following command:

```
git pull
```

If you would like an introduction to **git**, refer to the following URL:

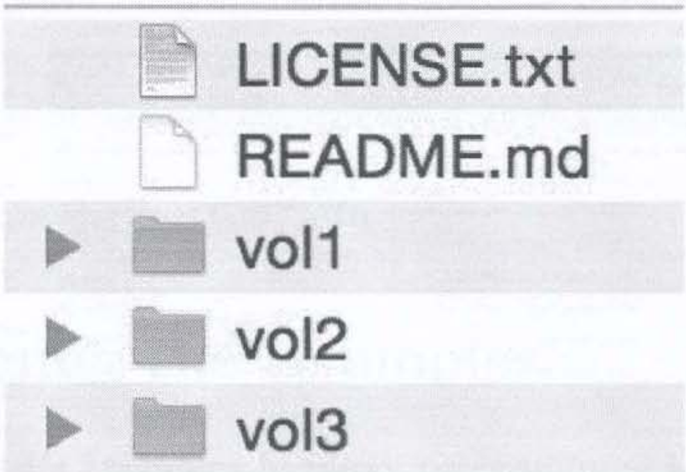
http://git-scm.com/docs/gittutorial

A.4 Example Contents

The entire *Artificial Intelligence for Humans* series is contained in one download that is a zip file.

Once you open the examples file, you will see the contents in Figure A.2:

Figure A.2: Examples Download



The license file describes the license for the book examples. All of the examples for this series are released under the Apache v2.0 license, a free and open-source software (FOSS) license. In other words, we do retain a copyright to the files. However, you can freely reuse these files in both commercial and non-commercial projects without further permission.

Although the book source code is provided free, the book text is not provided free. These books are commercial products that we sell through a variety of channels. Consequently, you may not redistribute the actual books. This restriction includes the PDF, MOBI, EPUB and any other format of the book. However, we provide all books in DRM-free form. We appreciate your support of this policy because it contributes to the future growth of these books.

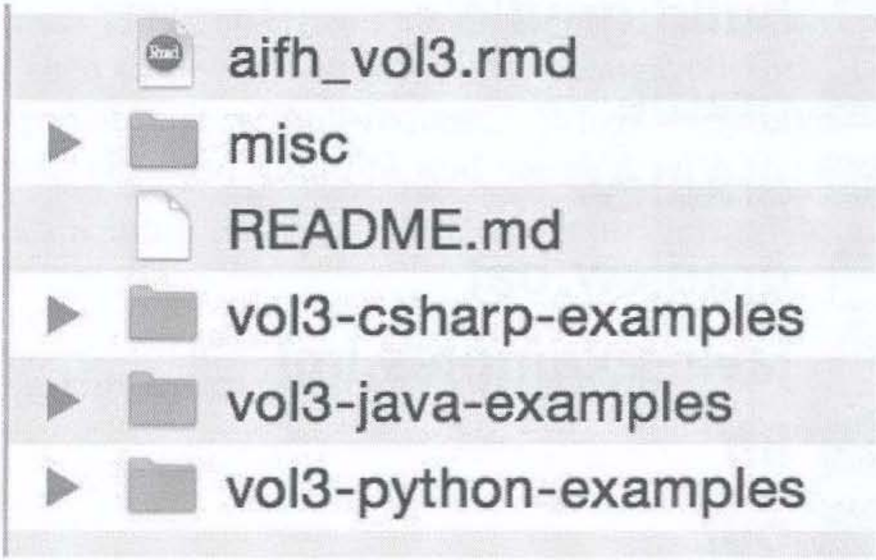
The download also includes a README file. The README.md is a "markdown" file that contains images and formatting. This file can be read either as a standard text file or in a markdown viewer. The GitHub browser automatically formats MD files. For more information on MD files, refer to the following URL:

https://help.github.com/articles/github-flavored-markdown

You will find a README file in many folders of the book's examples. The README file in the examples root (seen above) has information about the book series.

You will also notice the individual volume folders in the download. These are named vol1, vol2, vol3, etc. You may not see all of the volumes in the download because they have not yet been written. All of the volumes have the same format. For example, if you open Volume 3, you will see the contents listed in Figure A.3. Other volumes will have a similar layout, depending on the languages that are added.

Figure A.3: Inside Volume 3 (other volumes have same structure)



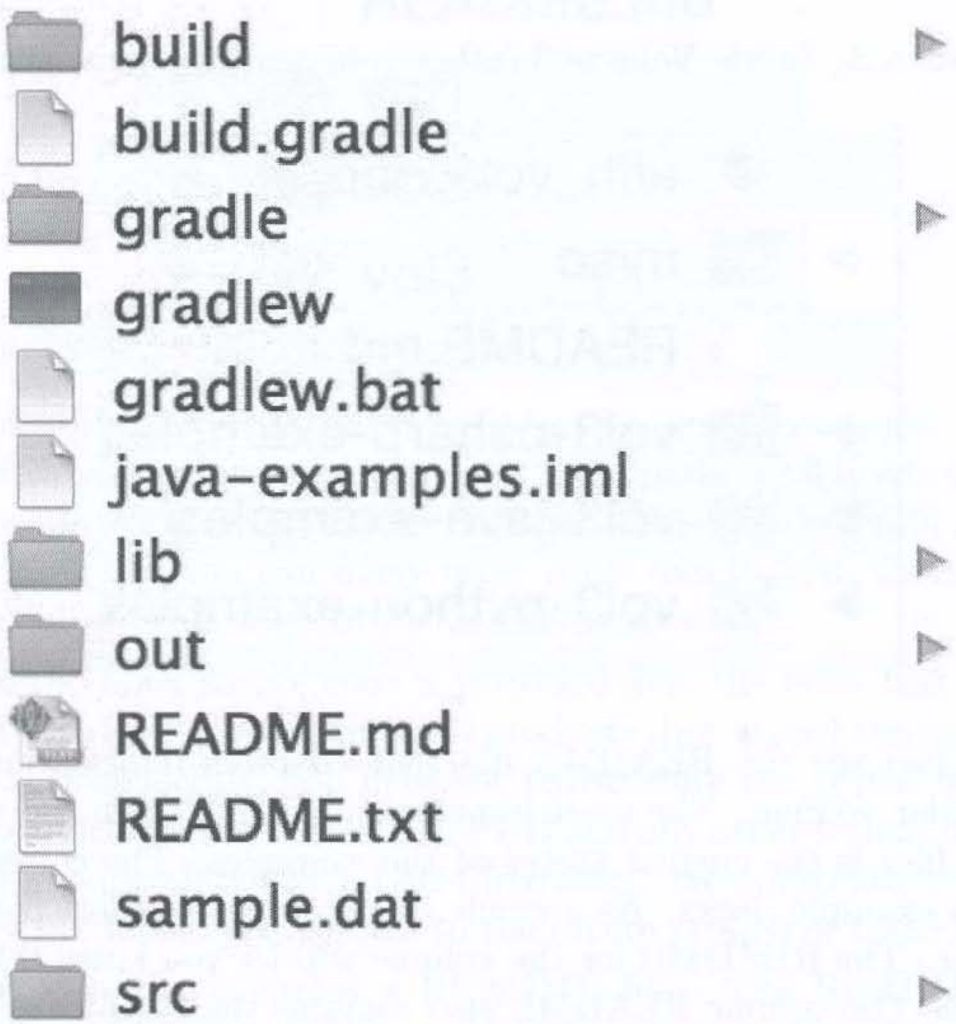
Again, you see the README file that contains information unique to this particular volume. The most important information in the volume level README files is the current status of the examples. The community often contributes example packs. As a result, some of the example packs may not be complete. The README for the volume will let you know this important information. The volume README also contains the FAQ for a volume.

You should also see a file named "aifh_vol3.RMD". This file contains the R markdown source code that we used to create many charts in the book. We produced nearly all the graphs and charts in the book with the R programming language. The file ultimately allows you to see the equations behind the pictures. Nevertheless, we do not translate this file to other programming languages. We utilize R simply for the production of the book. If we used another language, like Python, to produce some of the charts, you would see a "charts.py" along with the R code.

Additionally, the volume currently has examples for C#, Java, and Python. However, you may see that we add other languages. So, always check the README file for the latest information on language translations.

Figure A.4 shows the contents of a typical language pack:

Figure A.4: The Java Language Pack



Pay attention to the README files. The README files in a language folder are important because you will find information about the Java examples. If you have difficulty using the book's examples with a particular language, the README file should be your first step to solving the problem. The other files in the above image are all unique to Java. The README file describes these files in much greater detail.

A.5 Contributing to the Project

If you would like to translate the examples to a new language or if you have found an error in the book, you can help. Fork the project and push a commit revision to GitHub. We will credit you among the growing number of contributors.

The process begins with a fork. You create an account on GitHub and fork the AIFH project. This step creates a new project that has a copy of the AIFH files. You will then clone your new project through GitHub. Once you make your changes, you submit a "pull request." When we receive this request, we will evaluate your changes/additions and merge it with the main project.

You can find a more detailed article on contributing through GitHub at this URL:

https://help.github.com/articles/fork-a-repo

References

This section lists the reference materials for this book.

Ackley, H., Hinton, E., & Sejnowski, J. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, 147-169.

Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G. Bengio, Y. (2010, June). Theano: a CPU and GPU math expression compiler. In *Proceedings of the python for scientific computing conference (SciPy)*. (Oral Presentation)

Broomhead, D., & Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2, 321-355.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR, abs/1412.3555.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14 (2), 179-211.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36, 193-202.

Garey, M. R., & Johnson, D. S. (1990). *Computers and intractability; a guide to the theory of np-completeness.* New York, NY, USA: W. H. Freeman & Co.

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In G. J. Gordon, D. B. Dunson, & M. Dudk (Eds.), *Aistats* (Vol. 15, p. 315-323). JMLR.org.

Hebb, D. (2002). *The organization of behavior: a neuropsychological theory.* Mahwah N.J.: L. Erlbaum Associates.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, abs/1207.0580 .

Hopfield, J. J. (1988). Neurocomputing: Foundations of research. In J. A. Anderson & E. Rosenfeld (Eds.), (pp. 457-464). Cambridge, MA, USA: MIT Press.

Hopfield, J. J., & Tank, D. W. (1985). "Neural" computation of decisions in optimization problems. *Biological Cybernetics*, 52, 141-152.

Hornik, K. (1991, March). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4 (2), 251-257.

Jacobs, R. A. (1988). Increased rates of convergence through learning rate adaptation. Neural Networks, 1 (4), 295-307.

Jacobs, R., & Jordan, M. (1993, Mar). Learning piecewise control strategies in a modular neural network architecture. IEEE Transactions on Systems, Man and Cybernetics, 23 (2), 337-345.

Jordan, M. I. (1986). *Serial order: A parallel distributed processing approach* (Tech. Rep. No. ICS Report 8604). Institute for Cognitive Science, University of California, San Diego.

Kalman, B., & Kwasny, S. (1992, Jun). Why TANH: choosing a sigmoidal function. In Neural networks, 1992. *IJCNN, International Joint Conference on Neural Networks* (Vol. 4, p. 578-581 vol.4).

Kamiyama, N., Iijima, N., Taguchi, A., Mitsui, H., Yoshida, Y., & Sone, M. (1992, Nov). Tuning of learning rate and momentum on back-propagation. In *Singapore ICCS/ISITA '92. 'Communications on the move'* (p. 528-532, vol.2).

Keogh, E., Chu, S., Hart, D., & Pazzani, M. (1993). Segmenting time series: A survey and novel approach. In an edited volume, *data mining in time series databases.* Published by World Scientific Publishing Company (pp. 1-22).

Kohonen, T. (1988). Neurocomputing: Foundations of research. In J. A. Anderson & E. Rosenfeld (Eds.), (pp. 509-521). Cambridge, MA, USA: MIT Press.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (n.d.). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (p. 2012).

LeCun, Y., Bottou, L., Bengio, Y., & Haner, P. (1998). Gradient-based learning applied to document recognition. In *Proceedings of the IEEE* (pp.2278-2324).

Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. In International conference on machine learning (*ICML*).

van der Maaten, L., & Hinton, G. (n.d.). Visualizing high-dimensional data using t-SNE. *Journal of Machine Learning Research (JMLR)*, 9, 2579-2605.

Marquardt, D. (1963). An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11 (2), 431-441.

Matviykiv, O., & Faitas, O. (2012). Data classification of spectrum analysis using neural network. *Lviv Polytechnic National University.*

McCulloch, W., & Pitts, W. (1943, December 21). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biology*, 5 (4), 115-133.

Mozer, M. C. (1995). Backpropagation. In Y. Chauvin & D. E. Rumelhart (Eds.), (pp. 137{169). Hillsdale, NJ, USA: L. Erlbaum Associates Inc.

Nesterov, Y. (2004). *Introductory lectures on convex optimization: a basic course.* Kluwer Academic Publishers.

Ng, A. Y. (2004). Feature selection, l1 vs. l2 regularization, and rotational invariance. *In Proceedings of the twenty first international conference on machine learning* (pp. 78-). New York, NY, USA: ACM.

Neal, R. M. (1992, July). Connectionist learning of belief networks. *Artificial Intelligence*, 56 (1), 71-113.

Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In *IEEE international conference on neural networks* (pp. 586-591).

Robinson, A. J., & Fallside, F. (1987). *The utility driven dynamic error propagation network* (Tech. Rep. No. CUED/F-INFENG/TR.1). Cambridge: Cambridge University Engineering Department.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). Neurocomputing: Foundations of research. In J. A. Anderson & E. Rosenfeld (Eds.), (pp.696-699). Cambridge, MA, USA: MIT Press.

Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In *Proceedings of the 2012 IEEE conference on computer vision and pattern recognition (cvpr)* (pp. 3642-3649). Washington, DC, USA: IEEE Computer Society.

Sjberg, J., Zhang, Q., Ljung, L., Benveniste, A., Deylon, B., yves Glorennec, P., Juditsky, A. (1995). Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31, 1691-1724.

Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In F. Pereira, C. Burges, L. Bottou, & K. Weinberger (Eds.), *Advances in neural information processing systems 25* (pp. 2951{2959). Curran Associates, Inc.

Stanley, K. O., & Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10 (2), 99-127.

Stanley, K. O., DAmbrosio, D. B., & Gauci, J. (2009, April). A hypercubebased encoding for evolving large-scale neural networks. *Artificial Life*, 15 (2), 185-212.

Teh, Y. W., & Hinton, G. E. (2000). Rate-coded restricted Boltzmann machines for face recognition. In T. K. Leen, T. G. Dietterich, & V. Tresp (Eds.), *Nips* (p. 908-914). MIT Press.

Werbos, P. J. (1988). Generalization of backpropagation with application to a recurrent gas market model. *Neural Networks*, 1.

Zeiler, M. D., Ranzato, M., Monga, R., Mao, M. Z., Yang, K., Le, Q. V., Hinton, G. E. (2013). On rectified linear units for speech processing. In *ICASSP* (p. 3517-3521). IEEE.

Index

Artificial Intelligence for Humans

Volume 3: Deep Learning and Neural Networks
First Edition

Neural networks have been a mainstay of artificial intelligence since its earliest days. Now, exciting new technologies such as deep learn ing and convolution are taking neural networks in bold new direction In this book, we will demonstrate the neural networks in a variety of real-world tasks such as image recognition and data science. We examine current neural network technologies, including ReLU activatio stochastic gradient descent, cross-entropy, regularization, dropout, and visualization.

Artificial Intelligence for Humans is a book series meant to teach AI to those without an extensive mathematical background. The reader needs only a knowledge of basic college algebra or computer programming—anything more complicated than that is thoroughly explained. Every chapter also includes a programming example. Exar ples for this volume are provided in Java, C#, and Python. Other languages are planned.

Jeff Heaton is a data scientist, PhD student and indy publisher Specializing in Java, C#, C/C++, Python and R, he is an active technology blogger, open source contributor, and author of more than ten books. His areas of expertise include predictive modeling, data mining, big data, business intelligence, and arti cial intelligence. Jeff holds a Master's Degree in Information Ma agement from Washington University, is a senior member of th IEEE and a PhD student at Nova Southeastern University in cor puter science. He is the lead developer for the Encog Machine Learning Framework open source project and a fellow of the Life Management Institute (FLMI).

ISBN 9781505714340
9 781505 714340
90