interpretnn: Interpreting feedforward neural networks as statistical models

Andrew McInerney[1]

@amcinerney_

@andrew-mcinerney

andrew.mcinerney@ul.ie

Kevin Burke[1]

kevinburke.ie

[1] *Department of Mathematics & Statistics, University of Limerick*

Introduction

- Many neural network R packages available: **nnet** (Ripley and Venables, 2022), **neuralnet** (Fritsch, Guenther, and Wright, 2019), **keras** (Allaire and Chollet, 2023), and **torch** (Falbel and Luraschi, 2023).
- Goal of our **interpretnn** package: Allow for more useful and insightful statistical-based methods and outputs.
- We embed neural networks within likelihood estimation, providing model selection and significance testing.
- This bridges the gap between the explainability and flexibility of neural networks.

Installation

- You can install the development version of **interpretnn** from GitHub with:

```
# install.packages("devtools")
devtools::install_github(
  "andrew-mcinerney/interpretnn"
  )
```

Implementation

- Example: Boston Housing dataset.
- First, we fit the data using the **nnet** package.

```
library(interpretnn)
library(nnet)
# set.seed(100)
nn <- nnet(medv ~ ., data = Boston,
           size = 2, trace = FALSE,
           linout = TRUE, maxit = 1000)
```

- Then, convert this to an `"interpretnn"` object using the `interpretnn()` function.

```
intnn <- interpretnn(nn, data = Boston)
```

Model Summary

- Now, with the `"interpretnn"` object, `summary()` produces a statistically-based model summary.

```
summary(intnn)
```

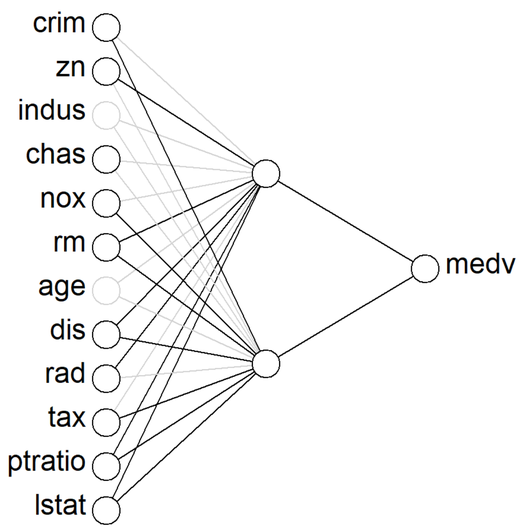
```
[...]
Number of input nodes: 12
Number of hidden nodes: 2

BIC: 606.4537

Coefficients:
                          Weights |       X^2     Pr(> X^2)
    crim      (-0.14, -0.52**) |  15.8610 3.60e-04 ***
      zn       (0.13**, 10.13) |   7.1319 2.83e-02 *
   indus            (-0.03, 0) |   0.1082 9.47e-01
    chas         (0.06., 0.12) |   8.1292 1.72e-02 *
     nox      (0.15, -1.42***) |  16.2004 3.03e-04 ***
      rm  (0.74***, -0.88***) | 102.7293 0.00e+00 ***
     age         (-0.05, -0.2) |   1.5472 4.61e-01
     dis (-0.35***, -1.85***) |  39.1672 3.13e-09 ***
     rad       (0.99***, 0.31) |  41.5644 9.43e-10 ***
     tax      (-0.13, -1.14**) |  10.4014 5.51e-03 **
 ptratio   (-0.16**, -0.66**) |  19.4630 5.94e-05 ***
   lstat (-1.38***, -0.69***) |  59.5259 1.19e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
0.1 ' ' 1
```

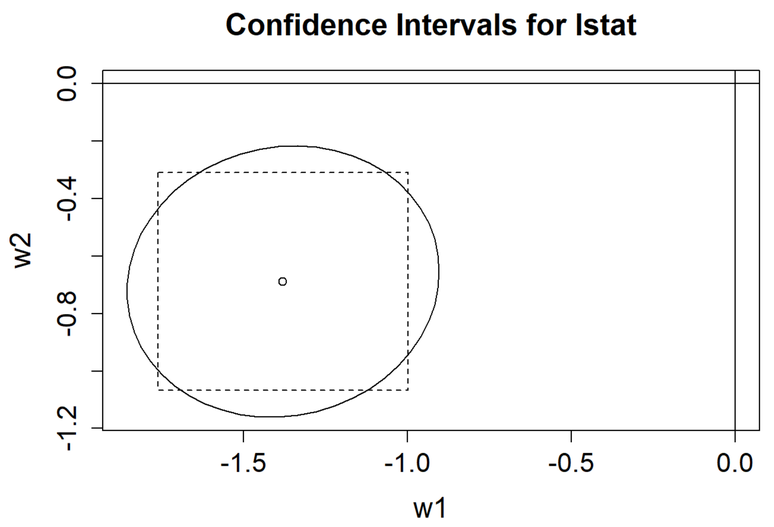
- `plotnn()` visualises the results of single- and multi-parameter Wald tests overlaid on the network architecture.
- By default, `alpha = 0.05`, where the weights are coloured black if they are significant, and are grey otherwise (i.e., insignificant).
- The intercept terms can be displayed by setting `intercept = TRUE` (default: `FALSE`).

```
plotnn(intnn)
```



- `plotci()` visualises the single- and joint-parameter Wald (1 - α)100% confidence intervals and ellipses, respectively, for the input-to-hidden-layer weights for each covariate.
- The `which` argument chooses a particular covariate as the subject of this plot (the default value, `NULL`, produces a plot for each covariate).

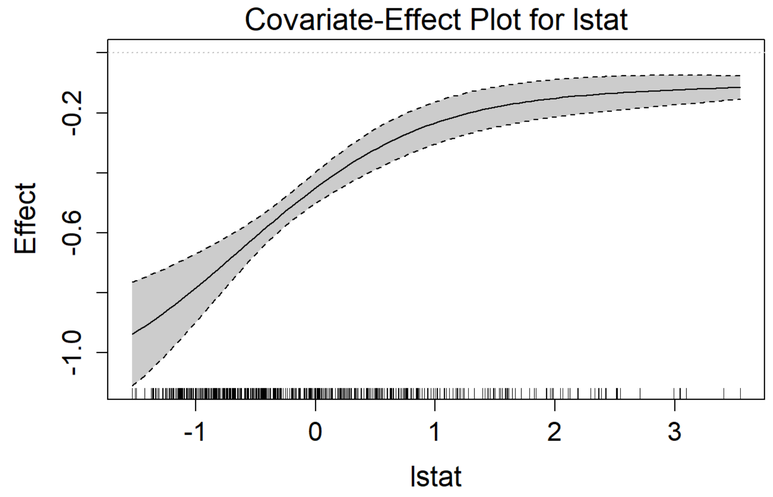
```
plotci(intnn, which = 12)
```



Covariate-Effect Plots

- `plot()` is used to display the covariate effects.
- To visualise the associated uncertainty, the `conf_int` argument can be set to `TRUE`.
- As before, `which` chooses a particular covariate as the subject of this plot (default: `NULL` produces a plot for each covariate).

```
plot(intnn, which = 12,
     conf_int = TRUE)
```



Acknowledgments

This work has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under Grant Number SFI 18/CRT/6049.

References

McInerney, A., & Burke, K. (2022a). *Interpretnn: Interpreting feedforward neural networks as statistical models*. Retrieved from https://github.com/andrew-mcinerney/interpretnn

McInerney, A., & Burke, K. (2022b). *Selectnn: A statistically-based approach to neural network model selection*. Retrieved from https://github.com/andrew-mcinerney/selectnn

McInerney, A., & Burke, K. (2022c). A statistically-based approach to feedforward neural network model selection. *arXiv preprint arXiv:2207.04248*.

McInerney, A., & Burke, K. (2023). Interpreting feedforward neural networks as statistical models. *In Preparation*.

Centre for Research Training | Science Foundation Ireland sfi For what's next

SFI Centre for Research Training in Foundations of Data Science

UNIVERSITY OF LIMERICK OLLSCOIL LUIMNIGH