Submitted to the Annals of Applied Statistics

DO FEEDFORWARD, DENSE NEURAL NETWORKS PREDICT A COUNTRY'S OBESITY PREVALENCE MORE ACCURATELY THAN ADDITIVE MODELS?

BY EDRIC SVARTE [1]

[1] *University of Waterloo, esvarteb@uwaterloo.ca*

We propose a model that predicts a nation's obesity prevalence. We compare the predictive performance of two model classes: neural networks and additive models. Implications and possible extensions are briefly discussed.

1. Introduction. Obesity is a pressing public health issue. An individual is obese if his BMI exceeds thirty[1] (Ritchie and Roser, 2017). A country's obesity prevalence is the percentage of adults who are obese (but not overweight). A predictive model of obesity would allow policymakers and health professionals to assess which countries are at risk of an obesity epidemic. Within this context, we compare the predictive performance of additive models (AMs) and feedforward, dense neural networks (NNs). *Feedforward* suggests data only travel forward: from input to output layer. *Dense* implies that neurons in adjacent layers are fully-connected pairwise.

Additive models are a class of flexible yet interpretable models. Despite their attractiveness, data scientists and machine learning engineers neglect AMs (Larsen, 2015). These practitioners prefer neural networks. These models have an impressive ability to "learn" complex relationships between variables. Unfortunately, these models are notoriously difficult to "train" (Nielsen, 2015), and generally uninterpretable. Indeed, NNs are often unfit for use in business and medicine (Agarwal et al., 2021).

In summary, we examine which model class more accurately predicts a nation's obesity prevalence. A model is *accurate* if its prediction error is relatively low.

1.1. *Related Literature.* Most of the literature on our current topic compares the performance of NNs to generalized additive models (GAMs). For instance, Zhou and Zhang (2022) attempt to predict the impact resistance of X80 pipelines. The authors propose a dense NN with three neurons, and one hidden layer. Unfortunately, this is an elementary neural network. Despite having only five input variables, it may be valuable to experiment with much deeper, wider networks. Finally, the authors show that the chosen NN is roughly three times as accurate as the chosen GAM.

Similarly, Papoila et al. (2013) compare GAMs, NNs and GLMs. The authors wish to predict the survival probability of critically ill hospital patients. Crucially, these classifiers[2] must be calibrated. For instance, suppose a model predicts that a patient has a 70% chance of survival. This patient should survive seven times out of ten if the model is calibrated. Unfortunately, modern neural networks tend to be poorly calibrated (Guo et al., 2017). Interestingly, Papoila et al. (2013) obtain similar results. In particular, they find that GAMs are better calibrated than NNs and GLMs. Fortunately, modern techniques, such as temperature scaling, can improve NN calibration. This process involves dividing predicted logits by a

Keywords and phrases: Obesity, additive models, neural networks.

[1] Some countries may define obesity otherwise. However, we use the WHO definition to be consistent with those who collected the data.

[2] Functions that handle categorical target variables. In this case, whether a patient will survive.

learned parameter. This simple method reduces NN overconfidence; the tendency to assign a large probability to the incorrect class (Guo et al., 2017).

On the other hand, Agarwal et al. (2021) propose a class of "glass-box" models: neural additive models (NAMs). Intriguingly, each input variable has its own neural network. These NNs are independent but are trained together. Finally, outputs are combined as in a GAM. The authors find that NAMs are nearly as accurate as NNs, though NAMs provide intelligible results. This approach seems promising.

2. Data.

2.1. *Sources and Exploratory Analysis.* All data are from ourworldindata.com, an online data repository. We combined four data sets from this source (all from 2013). The first data set provides the target variable: the obesity prevalence in 202 countries (Ritchie and Roser, 2017). The second provides four covariates, and details mean daily per capita calorie intake by macronutrient[3] in 173 countries (Ritchie, Rosado and Roser, 2017). The third details the average per capita daily calorie intake by food group[4] in 170 countries (Ritchie, Rosado and Roser, 2017). However, only one column (variable) was of interest: alcohol intake. Finally, the last data set details the per capita GDP of 183 countries (Roser, 2013). All four data sets were merged, resulting in a sample of $n = 159$ observations. There was one missing value: the mean daily alcohol intake in the United Arab Emirates (UAE). This data point (country) was removed. We proceed with a sample of $n = 158$ observations.

As stated, the obesity prevalence is the continuous target variable. The remaining six variables are continuous predictors. We perform a sample-split before attempting any sort of analysis. We create a training set, $\mathcal{T}$, and a hold-out test set, $\mathcal{H}$ (80/20 split). The test set will not be used for exploratory analysis or model selection. Indeed, $\mathcal{H}$ has a single purpose, which we describe in the next section (3.1).

The mean obesity prevalence was found to be $\hat{\mu} = 17$. That is, 17% of individuals are obese in a typical country. A 95% double-bootstrap confidence interval for the true mean is given by $\hat{I}^{\mu}_{95} = [15.4,\ 18.5]$; there is little uncertainty surrounding the estimate of the mean.

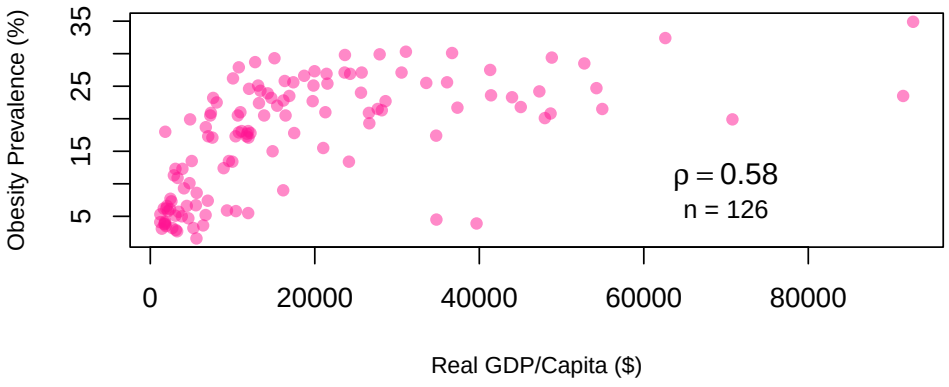

FIGURE 1. *A Non-Linear Relationship in the Training Data*

[3]Macronutrients include carbohydrates, fats, and protein. Protein is further separated by source: plant or animal. Hence there are four variables.

[4]For instance, dairy and eggs, pulses, starchy roots.

The plot above shows the obesity prevalence as a function of the per capita, real GDP. Clearly, the relationship is not linear on the entire domain. The other five variables likewise have non-linear relationships with the target; hence the need for flexible models.

3. Methods.

3.1. *Outline.* The analysis has two stages. In stage one (sections 4.1, 4.2), we build and evaluate several AMs and NNs (separately). We evaluate models on $\mathcal{T}$ using five-fold cross-validation, repeated ten times to stabilize the results. $K = 5$ is less computationally intensive than $K = 10$ or LOOCV; it is not feasible to fit a neural network n times per CV repetition. At the end of stage one, we use the CV results to select one AM and one NN. An *optimal* model is one that minimizes the cross-validation error, $MSPE_{CV}$. However, parsimony (simplicity) is also a consideration. In stage two (section 4.3), we use the two chosen models to make a final prediction using the test set features. We then compare the results.

3.2. *Neural Networks.* Here we do not perform variable selection. However, we must select hyperparameters. We detail our process below.

We use a grid search to find a locally optimal width/depth combination. A heuristic provides the grid upper bound for the width, w: the number of neurons should be (roughly) less than twice the input dimension (Ke and Liu, 2008). Selecting the depth, d, is more complicated, though two hidden layers suffice for most purposes.[5] However, we choose to experiment with every combination of width and depth from the sets $w = \{2,\ 4,\ 8,\ 16\}$, and $d = \{1,\ 2,\ 4\}$.

Next, we use two methods to prevent overfitting. First, we use L_2 regularization: we choose the regularization parameter from the set $\lambda_k = 10^{-k},\ k = [-4,\ 3] \subset \mathbb{Z}$ which minimizes $MSPE_{CV}$. Second, CV will reveal any overfit models. Note that we test λ values and the different width/depth combinations simultaneously (a 3D grid search). There are hence $n(w) \cdot n(d) \cdot n(\lambda) = 4 \cdot 3 \cdot 8 = 96$ neural networks to evaluate. Lastly, the number of epochs must be chosen to balance bias and variance. We choose to vary the number of epochs until $MSPE_{CV}$ stops decreasing.

Next, we attempt to accelerate training. First, we standardize input batches (subsets of $\mathcal{T}$) after every dense layer, before passing them to the activation function. We use a batch size of twenty-two since small batch sizes may improve generalization (Oyedotun, Papadopoulos and Aouada, 2022). Next, we use the ReLU activation function to enhance computational efficiency.[6] Lastly, we use stochastic gradient descent (SGD) to minimize the squared loss function. Note that we begin with a learning rate of 0.001 and divide it by ten every fifty epochs to avoid missing minima.

We use "early-stopping" to train the final NN after choosing hyperparameters. In particular, we split the training set, $\mathcal{T}$, into a smaller training set, $\mathcal{T}_{MINI}$, and a validation set, $\mathcal{V}$ (70/30 split). We train the model on $\mathcal{T}_{MINI}$ until the error on $\mathcal{V}$ stops decreasing. This technique yields acceptable results despite being slightly crude (Lodwich, Rangoni and Breuel, 2009).

3.3. *Additive Models.* Here we have a variable selection problem, which we solve using a type of backward elimination. We begin with a model with all six predictors, and evaluate it via CV. We then remove variables one-by-one, and select the (five-variable) model that

[5] Since these networks can generate functions which are dense in the set of continuous functions mapping X to Y (Stathakis, 2009).

[6] The ReLU's derivative is either zero or one, hence simple to calculate.

minimizes $MSPE_{CV}$. We continue this greedy search until one variable remains (the stopping rule). Crucially, we remove one variable at each step even if doing so does not decrease $MSPE_{CV}$. This design allows us to explore more of the search space, and may help us overcome local minima. Hence each step yields a (locally) optimal model for a given number of variables. We then choose one of six models; the one which minimizes $MSPE_{CV}$.

Next, we select hyperparameters. Fortunately, the *gam()* function from the MGCV package optimizes these values via generalized CV. We let this function choose the number and placement of knots, as well as the regularization parameters.

4. Results.

4.1. *Optimal Neural Network.* The table below details $MSPE_{CV}$ values for various NNs, given an optimal tuning parameter, λ^*, and an optimal number of epochs, E^*.

TABLE 1
Neural Network Cross-Validation Results

	Width (w)			
Depth (d)	2	4	8	16
1	55.4	54.0	55.1	55.7
2	35.2	34.5	34.1	34.8
4	37.3	37.0	37.7	37.7

It seems that two hidden layers minimizes CV error. In particular, the NN with $w = 8,\ d = 2$ achieved the lowest error: $MSPE_{CV} = 34.1$. Similarly, the NN with $w = 4,\ d = 2$ achieved an error of $MSPE_{CV} = 34.5$. However, the former is significantly more complicated than the latter (137 vs. 53 parameters). We apply the parsimony principle, and deem $w^* = 4,\ d^* = 2$ the optimal width/depth combination. This model's error was minimized by training for $E^* = 250$ epochs on each fold, and $\lambda^* = 10$.

Next, we trained the optimal NN on $\mathcal{T}_{MINI}$. The validation error stopped decreasing after $E = 40$ epochs. We now train the chosen neural network on $\mathcal{T}$ with $E^* = 40$ epochs, and report the results in section 4.3.

4.2. *Optimal Additive Model.* The table below details $MSPE_{CV}$ values for the optimal AM at each step of the greedy search (the algorithm proceeds from top to bottom).

TABLE 2
Additive Model Cross-Validation Results

	Variables Included						
Model	FatCals	CarbsCals	AnimalProCals	PlantProCals	AlcoholCals	GDP	MSPE
1	1	1	1	1	1	1	37.4
2	1	1	1	0	1	1	36.6
3	1	1	0	0	1	1	34.0
4	1	1	0	0	1	0	35.8
5	1	1	0	0	0	0	34.5
6	0	1	0	0	0	0	36.2

Model 3 and 5 minimize $MSPE_{CV}$. However, model 5 includes two variables, while model 3 includes four. We apply the parsimony principle and select model 5.

4.3. *Optimal Model.* We used the features from the hold-out test set to make a final prediction. Using the chosen NN, we find that $MSPE_{TEST} \approx 41$ and $\hat{I}_{95}^{MSPE_{TEST}} = [22.1,\ 72.9]$. Hence the model is imprecise; the error magnitude is quite variable. On the other hand, the chosen AM's error was found to be $MSPE_{TEST} \approx 19$, with corresponding interval estimate $\hat{I}_{95}^{MSPE_{TEST}} = [10.4,\ 35.2]$. Thus the AM is more than twice as accurate as the NN. Moreover, there is less uncertainty surrounding this estimate. Based on these results, we propose an additive model. Recall that this model has two predictors, namely the mean daily consumption of fats and carbohydrates. We explore this model in the next section.

4.4. *Interpretation.* Here we examine one of the model's smooth functions.

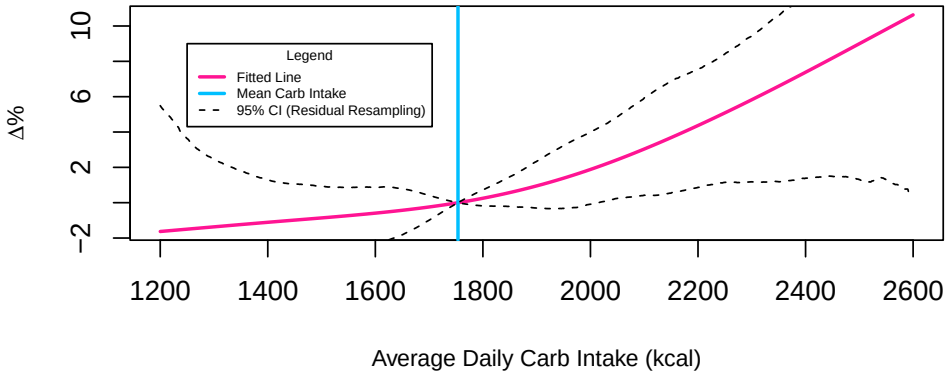

FIGURE 2. *Marginal Effect of Carbohydrate Intake on Obesity Prevalence*

Note that f_{CARB} is equal to zero at the mean daily carbohydrate intake (roughly 1750 calories). Interestingly, this function is convex: carbohydrate intake has an increasing marginal effect on the rate of obesity. Suppose a nation's citizens consume an average of 1750 carbohydrate calories a day. This country's obesity prevalence should rise by roughly four percent should consumption increase to 2200 calories.

5. Conclusion.

5.1. *Discussion.* First, all data are from 2013, and thus may be unrepresentative of the world today. For instance, GDP per capita tends to increase over time (Roser, 2013). Moreover, the joint distribution of the variables may have changed. However, future research may use these results as a starting point.

Next, we discuss a surprising finding: AMs outperformed NNs. Crucially, this result is likely due to "user error," and the inherent challenges of non-convex optimization. For instance, we used relatively simple and crude methods to select hyperparameters. Indeed, the analysis could be repeated using different techniques. For example, dropout could replace L_2 regularization, and the learning rate could cycle rather than decay.

Similarly, our AM selection strategy has its limitations. For example, we used a greedy algorithm to select features, and thus tried a fraction of all possible covariate subsets. Furthermore, we did not consider variable interactions. Perhaps the association between obesity and fat intake depends on real GDP. However, we appreciate AMs for their interpretability. Indeed, interactions may be tough to comprehend for those not well-versed in statistics or mathematics. On the other hand, modelling these interactions constitutes a possible extension. Despite these limitations, we obtained satisfactory results.

We should also mention that CV does not estimate a model's prediction error, but rather the generalization error averaged across all training sets. However, CV may still be useful in *comparing* models (Bates, Hastie and Tibshirani, 2023, 3).

We now briefly mention model assumptions. It is unclear whether the additivity and smoothness assumptions hold. As mentioned, however, these assumptions ensure interpretability and simplicity. We performed a brief residual analysis. The residuals do seem randomly scattered about the zero line. Next, both the Fligner-Killeen and Levene tests were non-significant. The normality assumption may be violated. For instance, both the Shapiro-Wilk and Jarque-Bera tests were significant, though the Anderson-Darling test was not.

Lastly, the proposed model should be tested on a new, independent test set. Indeed, we used our test set, $\mathcal{H}$, to compare the optimal NN to the optimal AM. In fact, the test set should only be used to provide a final estimate of the selected model's generalization error. However, the principal goal of this study was to compare two model classes. Our use of the test set was appropriate given this objective.

5.2. *Conclusion.* We used five-fold cross-validation, repeated ten times, to select one additive model and one neural network. Next, the predictive performance of both models was assessed on a holdout test set, $\mathcal{H}$, with $n = 32$ observations. We found that $MSPE_{NN} \approx 41$ and $MSPE_{AM} \approx 19$. The AM was found to be more precise and accurate than the NN. On average, the AM mistakenly predicts the obesity rate by roughly 4.4%. This model includes two covariates: the mean daily per capita consumption of fats and carbohydrates. We predict that countries whose inhabitants consume large amounts of these macronutrients will have higher rates of obesity. This finding has implications for health professionals and policymakers alike.

REFERENCES

AGARWAL, R., MELNICK, L., FROSST, N., ZHANG, X., LENGERICH, B., CARUANA, R. and HINTON, G. E. (2021). Neural additive models: Interpretable machine learning with neural nets. *Advances in Neural Information Processing Systems* **34** 4699–4711.

BATES, S., HASTIE, T. and TIBSHIRANI, R. (2023). Cross-validation: what does it estimate and how well does it do it? *Journal of the American Statistical Association* 1–12.

GUO, C., PLEISS, G., SUN, Y. and WEINBERGER, K. Q. (2017). On calibration of modern neural networks. In *International conference on machine learning* 1321–1330. PMLR.

KE, J. and LIU, X. (2008). Empirical analysis of optimal hidden neurons in neural network modeling for stock prediction. In *2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application* **2** 828–832. IEEE.

LARSEN, K. (2015). GAM: The Predictive Modeling Silver Bullet. https://multithreaded.stitchfix.com/blog/2015/07/30/gam/. Accessed: 2023-06-13.

LODWICH, A., RANGONI, Y. and BREUEL, T. (2009). Evaluation of robustness and performance of early stopping rules with multi layer perceptrons. In *2009 international joint conference on Neural Networks* 1877–1884. IEEE.

NIELSEN, M. A. (2015). *Neural networks and deep learning* **25**. Determination press San Francisco, CA, USA.

OYEDOTUN, O. K., PAPADOPOULOS, K. and AOUADA, D. (2022). A new perspective for understanding generalization gap of deep neural networks trained with large batch sizes. *Applied Intelligence* 1–17.

PAPOILA, A. L., ROCHA, C., GERALDES, C. and XUFRE, P. (2013). Generalized linear models, generalized additive models and neural networks: comparative study in medical applications. In *Advances in regression, survival analysis, extreme values, Markov processes and other statistical applications* 317–324. Springer.

RITCHIE, H., ROSADO, P. and ROSER, M. (2017). Diet Compositions. *Our World in Data*. https://ourworldindata.org/diet-compositions.

RITCHIE, H. and ROSER, M. (2017). Obesity. *Our World in Data*. https://ourworldindata.org/obesity.

ROSER, M. (2013). Economic Growth. *Our World in Data*. https://ourworldindata.org/economic-growth.

STATHAKIS, D. (2009). How many hidden layers and nodes? *International Journal of Remote Sensing* **30** 2133–2147.

ZHOU, Y. and ZHANG, S. (2022). Prediction of rupture and perforation limits of pressurised X80 pipelines using BP neural networks and generalised additive models. *Ocean Engineering* **259** 111839.