Value-at-risk Forecasting via Sieves

some guidance for neural nets

Philipp Ratz[a]

[a]Université du Québec à Montréal (UQAM), Montréal QC, Canada

arXiv: 2205.07101

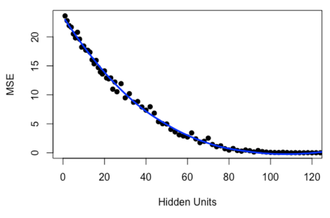
Online Appendix
Contains mathematical derivations and details on simulations

Introduction

Take almost any modern application involving ANNs, chances are it is modelled using the **ReLU activation function**. The main reason why ReLU established itself as the go-to standard among a host of activation functions is that it does not suffer from the **vanishing gradient problem**. Whereas the success of ANNs in areas such as **image and text recognition** is ubiquitous, the evidence of their success in areas that use tabular data has been mixed at best. For example, in a **study on real time series data** analysed by Makridakis et al. (2020) ANNs sometimes **performed worse simple benchmarks**. This raises the question why this might be the case. In what follows we first **extend theoretical results** on the bound of convergence to include ReLU activation functions and then compare these bounds to other nonparametric estimators. Simulations are then presented to investigate **finite sample properties**. Finally, an application to estimate the **Value-at-Risk** (VaR) presents how the derived theoretical guidelines can be used in real world problems.

The sieve method

To obtain **convergence results** we follow previous literature and consider the ANN as a **sieve estimator**, which in essence means having more complex models with more data. For neural networks sieve estimation has a natural interpretation - if we have more data we can express more **complex functions** without just fitting noise, and we control the **model complexity** by the **number of hidden units**. The intuition is quite simple, assume $\hat{\phi}(\cdot)$ is the $\arg\min$ for some loss function, in our example the least squares. Instead of **considering** $\phi(\cdot)$ to be located in some **fixed (and finite) parameter space**, the method of sieves considers the $\arg\min$ merely the **solution in some subspace** Θ_n which is contained in the (infinite) parameter space Θ. The dimension of Θ_n is then **dependent on the size of the data** n, non decreasing in n and in the limit is contained in Θ (ie. $\Theta_n \subseteq \Theta, \forall n$). Formally, we can consider a sieve estimator $\hat{\theta}$ as:



Model flexibility and number of hidden units

$$\hat{\phi}(\cdot, \hat{\theta}) = \arg\min_{\theta \in \Theta_n} \frac{1}{n} \sum_{t=1}^{n} (y_t - \psi_n(x_t, \theta))^2 \,,$$

for some target y_t and some input x_t. This formulation might seem superfluous but it enables us to **derive bounds on the convergence speed**. Specifically, if we consider the simplest case of an ANN, the **multilayer perceptron with a single hidden layer** of the form:

$$f(x, \theta) = \gamma_0 + \sum_{u=1}^{u_n} \gamma_u \text{ReLU}(\boldsymbol{x}^\top \mu_u)$$

we then show that the approximation rate of the estimator $\hat{\theta}_n$ to the true (possibly infinite dimensional) parameter θ_0 can be **bounded above**. This has already been done before, but in our Article, we show that **the same bounds also apply to ReLU-sieves**

Convergence and implications

For the estimator with **ReLU activation function** we get our main theoretical result that holds for either i.i.d or β-mixing time-series data with **dimensionality** d and **sample size** n.

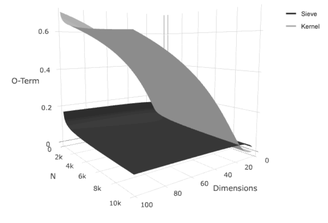
Upper bound on convergence

The upper bound on the convergence of the sieve parameter $\hat{\theta}_n$ towards the true parameter θ_0 can be expressed as:

$$||\hat{\theta}_n - \theta_0|| = \mathcal{O}\left([n/\log(n)]^{\frac{-(1+\frac{2}{d+1})}{4(1+\frac{1}{1+d})}} \right)$$

Where we assume that the function ϕ that we try to approximate is square integrable. Further, note that this rate is exactly the same that was derived in earlier work such as Chen et al. (2001) but for activation functions that suffer from the *vanishing gradient* problem.

With the convergence term above we can then **compare the sensitivity of the convergence** of ANNs with other nonparametric estimators. As the rate derived is an **upper bound using the simplest form** of an ANN but there is (at least heuristic) **evidence** that even for simple functions **multi-layer ANNs** might improve the fit (example in the appendix). Further, it seems that the expression inside the $\mathcal{O}$ term is **larger for low-dimensional problems** but a lot less sensitive to the dimensionality than, for example, kernel density estimators. As the rates are not directly comparable with finite samples (as they might be multiplied by an arbitrary constant) we investigate the properties via monte carlo simulations.



Comparison of $\mathcal{O}$-term

Monte Carlo Study

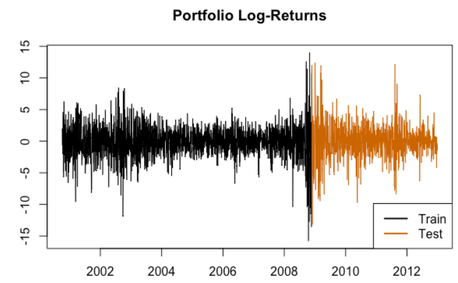
Here we try to approximate a **multi-parameter, highly nonlinear function** (follow QR-code for details) via ANN and kernel density estimation. To every function we successively add more noise terms to inflate the dimensionality of the problem without adding value to the predictions. **The predictive performances** are reported in Table 1. For **lower dimensional problems**, the local linear estimator outperforms the ANN but in **higher dimensional** or noise inflated scenarios the ANN starts to perform better in terms of MSPE.

		Relevant Predictors		
	Noise	2	5	15
Nonparametric (LL)	1	6.11*	18.31*	59.13
	10	28.98	46.48	79.30
	20	44.02	66.65	97.99
ANN	1	22.33	33.99	41.86*
	10	27.10*	36.85*	47.03*
	20	34.87*	43.27*	51.72*

Table 1: MSPE of ANN and KD estimators for varying complexity of functions. The star (*) denotes the lowest MSPE value across the models

Application to VaR forecasting

Finally we estimate the **VaR of a single stock and a portfolio** of stocks and commodities with competing model classes. The nonparametric estimation via sieves corresponds to a **time-varying intercept model** that is composed of both a lagged parametric part and nonparametric covariates. We model the quantiles directly using the ANN estimator with an adapted loss function. The form of the model is a **semi-parametric version of the CAViaR** model of Engle and Manganelli (2004) and takes the following form:



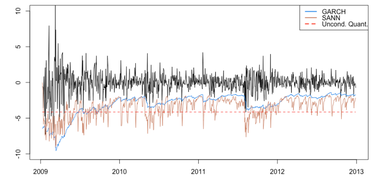
Log returns of the portfolio of stocks and commodities

$$VaR_t(\theta) = \sum_{j=1}^{l} \alpha_j VaR_{t-j}(\theta) + \phi(x_{t-1})$$

Linear models often seem apt during periods of **low volatility**, but having a model that can also **weather stormy seas** should be the goal when performing effective risk management. Therefore, we deliberately set the **test period during the stress period of the great recession**. We compare both parametric and nonparametric estimates for a stock and a portfolio, we find that that the use of the (more complicated) sieve estimators leads to **more precise outcomes in term of exceedances** as compared to a GARCH model, but also leads to significantly higher overall VaR values. Overall they **perform better for portfolios**.

	GARCH(2,1)	ANN	MGARCH(1,1)	MANN
Exceedances	14	11	23.02	10.56
Failures Test	0.23	0.75	0.52	0.02
Duration Test	0.15	0.01	0.12	0.06
Change $\int \text{VaR}_\alpha$	-27.7%	-13.8%	-25.0%	-6.1%
Coef. VaR_{t-1}		0.66		0.46
Std. Error		0.19		0.11

Table 2: *Predictive performance across univariate and multivariate models. For the tests of the multivariate models the fraction of rejected portfolios is reported (eg. 0.02 - 2% of portfolios were rejected by the test)*



Value at Risk forecast on test set

References

phi-ra.github.io/projects/sieve_estimation

Chen, X., Racine, J., and Swanson, N. (2001). Semiparametric arx neural-network models with an application to forecasting inflation. *IEEE Transactions on Neural Networks*, 12(4):674–683.

Engle, R. F. and Manganelli, S. (2004). Caviar: Conditional autoregressive value at risk by regression quantiles. *Journal of business & economic statistics*, 22(4):367–381.

Makridakis, S., Spiliotis, E., and Assimakopoulos, V. (2020). The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1):54–74.