# Deep Learning for Risk in Commercial Mortgage

Arka Prava Bandyopadhyay* Yildiray Yildirim†

March 29, 2019

## 1 Extented Abstract

We implement a deep neural network methodology of multi-period (monthly) risk for commercial mortgages and using a proprietary Trepp dataset. We use 84 loan-specific, property-specific, bond-specifc and deal-specific co-variates and 4 macro-economic variables for approximately 90,000 loans over a period of 21 years from 1998 to 2018. We classify the delinquency behavior of loans into 4 different states starting from Current/Performing (state 0) , Late/Non-Performing (state 1), Liquidation Proceedings (state 2), Final Resolution (state 3) and use hidden layers to capture the inherent information contained in the panel data. We see expotential boost in computational time by adding deeper layers in the neural network. However, the significant increase in accuracy of predictions is observed when the Deep Learning Model is compared with Naive Bayes, Random Forest, Support Vector Machine and Multinomial & Ordered Logistic Regression. We also extend the literature on frailty models Duffie (2009), where clustering of default could be captured only by linear unobserved time-fixed and loan-fixed effect. The ReLu activation function in the Deep Neural Network captures the highly non-linear interaction among co-variates, geographical cross-correlation and macro-economic factors at state level. These findings have significant implications for CMBS investors, rating agencies, and commercial property finance policymakers.

Our empirical results are based on a deep learning model for mortgage states over several periods with monthly frequency. The model harnesses the unprecedented size of our sample set and the extensive risk factors we investigate. The representation captures the nonlinear relationships between covatiates and mortgage state probabilities in the data. The model distinguishes between multiple states including current, 30, 60 & 90+ days past due, foreclosure, REO and prepaid. Our deep learning model for commercial mortgage state probabilities is a nonlinear extension of the multinomial/ordered logistic regression model. It can be viewed as a recursive formulation of logistic regression of nonlinear transformations of the explanatory variables.

We calculate the accuracy of prediction for 4 different delinquency states starting from Current/Performing (state 0) which includes "loans with payments not received but still in grace period or not yet due", Late/Non-Performing (state 1) which includes loans with "Late Payment

*PhD Student, Department of Real Estate, Baruch College, New York, NY 10010 USA, ArkaPrava.Bandyopadhyay@baruch.cuny.edu

†Department of Real Estate, Baruch College, New York, NY 10010 USA, Yildiray.Yildirim@baruch.cuny.edu

But Less Than 30 Days Delinquent", 30-days, 60-days, 90-days, 121+ days delinquent loans, Liquidation Proceedings (state 2) which implies "Foreclosure in process" and Final Resolution (state 3) which means REO. Loans transitioning between the states 2 & 3 are called "limbo" loans.

# 2 Data

We have monthly proprietary novel data set from 1998 to 2016 from Trepp. For the initial review, we use data from 2007 to 2009 (36 months) for 91767 loans for model fitting purposes, since going beyond this size requires access to GPU/Clouds, which we will adapt soon. Our final cleaned data has around 2.2 million observations of 48 covariates with a data size of nearly 1 GB.

From literature review, we list the variables used in previous CMBS papers, like An, Deng, and Gabriel (2009), Ambrose and Sanders (2003) and preclude the following key loan-specific variables: log(original balance), LTV, time of amortization, time to maturity, lockout, lockout expiration, corporate bond credit spread Titman, Tompaidis, and Tsyplakov (2005), yield curve, mortgage-treasury rate spread, 10 year risk-free rate volatility Downing, Stanton, and Wallace (2008) , multifamily REITs index volatility, region dummy, seasonal/quarter dummy, among others.

The local macro-variables capture the channel of potential contagion in which defaults in the same state might amplify the barely surviving loans. Such feedback mechanisms are well-documented by Agarwal, Ambrose, and Yildirim (2015), Anenberg and Kung (2014), Harding, Rosenblatt, and Yao (2009), Lin, Rosenblatt, and Yao (2009), Towe and Lawley (2013), and among others. Additionally, we included state-level quarterly GDP from 2005-2018, monthly historical unemployment data by state from 1976 to 2018, and historical interest rates of different maturities.

# 3 Deep Learning Model for CMBS

Deep Learning is a form of machine learning with multiple layers that learns multiple levels of representations for different levels of abstraction. It captures associations and discovers regularities within sets of patterns; it is suited for high volume, high dimensional data. It performs well when the relationships are dynamic or non-linear , when the standard regression models perform very poorly. No assumptions on normality, linearity, variable independence are needed.

The purpose for a deep learning model is bourne out of the need to have transparency and accountability. By the very nature of the deep learning model, we do not have to add interaction terms in the specification of the model, especially in the case of high dimensional data. The sequential layers embody highly non-linear and non-trivial interaction among the variables and capture several latent fundamental features in the process. The causal interpretation of the covariates both in default and prepayment calculations have not been explored in details. The broader impact could be traced out by improved allocation of credit and aid in policy design (macroprudential, bankruptcy, foreclosure, etc.).

With the provision of enough hidden units, a neural network can mimic continuous functions on closed and bounded sets really well Hornik (1991), vis-a-vis the product and division of

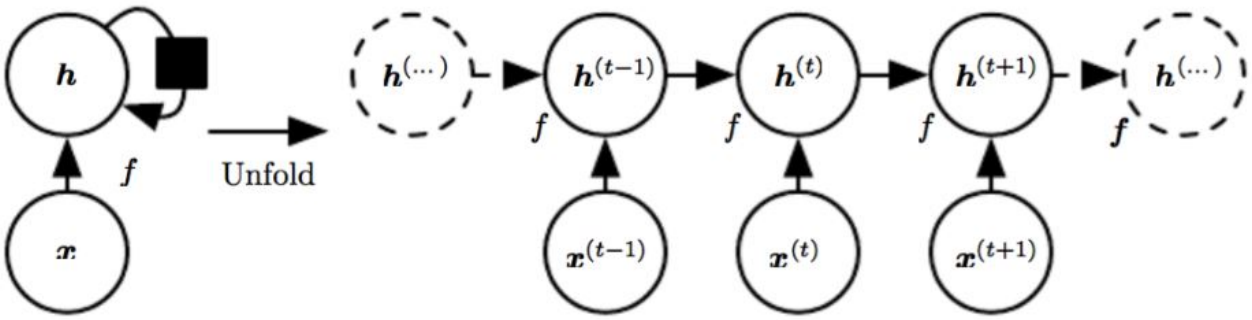



**Figure 1:** An recurrent network with no inputs

relevant features and their interactions. The advantage of more layers (as opposed to simply adding more units to existing layers) is that the later layers learn features of greater complexity by utilizing features of the lower layers as their inputs. Deep neural networks, with three or more hidden layers, t need exponentially fewer units than shallow networks or logistic regressions with basis functions; see Montufar, Pascanu, Cho, and Bengio (2014) and Goodfellow, Bengio, and Courville (2016).

## 3.1 Dynamic Deep Recurrent Neural Network

Recurrent Neural Networks (henceforth referred to as RNN) are neural networks that are prefered to Convolutional Neural Networks (CNN) for the purpose of processing sequential data $x^{(1)}, ..., x^{(\tau)}$ of variable length. Parameter sharing enables the extension of multilayer perceptrons (MLP) to RNNs. Using a time dependent parameter, sequence lengths not seen during training cannot be generalized. Nor can we translate the statistical robustness across different sequence lengths and also across time. Each member of the output is computed uniformly by the same update methodology applied to previous outputs, which results in parameter sharing vis-a-vis a very deep computational graph.

A computational Graph is a visualization for comprehending the unfolding of a recursive or recurrent computation, corresponding to a sequence of events, allowing for parameter sharing.

We consider a specification as in Goodfellow, Bengio, and Courville (2016):

$$h^{(t)} = g^{(t)}(x^{(t)}, x^{(t-1)}, x^{(t-2)}, ..., x^{(2)}, x^{(1)}) \tag{1}$$

Here, $h(.)$ represents the hidden layers, which are lossy summaries. Depending on the training criterion, the summary will keep certain aspects of the past more accurately than others. The extreme case is when one can recover the input sequence from $h^{(t)}$, e.g., in autoencoder frameworks. $g^{(t)}$ needs the entire history $(x^{(t)}, x^{(t-1)}, x^{(t-2)}, ..., x^{(2)}, x^{(1)})$ as input to produce the current state.

The recursive specification described below has two distinct advantages:

$$h^{(t)} = f(h^{(t-1)}, x^{(t)}; \theta) \tag{2}$$

1. Whatever be the sequence length, the learned model has the same input size, because it is specified in terms of transition between consecutive states, rather than in terms of variable-length history of states.

2. The same transition function $f(.)$ can be used with the same parameters at every time step.

Learning a single shared parameter model is the key to generalizing to sequence lengths not in the training set. This facilitates the estimation with way less training examples than 0ther methods not using parameter sharing.

Under certain regularity conditions, the estimators are consistent and also asymptotically normal (see Hornik, Stinchcombe, and White (1989), Sussmann (1992) and Albertini and Sontag (1993) where they study the identifiability). One can regularize DNN using optimal hyper-parameter tuning via cross-validation and "drop out" some sample values to reduce overfitting Srivastava, Hinton, Krizhevsky, Sutskever, and Salakhutdinov (2014).

# References


Agarwal, Sumit, Brent W. Ambrose, and Yildiray Yildirim, 2015, The Subprime Virus, *Real Estate Economics* 43, 891–915.

Albertini, Francesca, and Eduardo D. Sontag, 1993, For neural networks, function determines form, *Neural Networks* 6, 975 – 990.

Ambrose, Brent W., and Anthony B. Sanders, 2003, Commercial Mortgage-Backed Securities: Prepayment and Default, *The Journal of Real Estate Finance and Economics* 26, 179–196.

An, Xudong, Yongheng Deng, and Stuart A. Gabriel, 2009, Value Creation through Securitization: Evidence from the CMBS Market, *Journal of Real Estate Finance and Economics* 38, 302–326.

Anenberg, Elliot, and Edward Kung, 2014, Estimates of the Size and Source of Price Declines Due to Nearby Foreclosures, *American Economic Review* 104, 2527–51.

Downing, Christopher, Richard Stanton, and Nancy Wallace, 2008, Volatility, mortgage default, and CMBS subordination, *Working Paper*.

Duffie, Darell, 2009, Frailty Correlated Default, *Journal of Finance* LXIV, 34–52.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville, 2016, *Deep Learning*. The MIT Press.

Harding, John P., Eric Rosenblatt, and Vincent W. Yao, 2009, The contagion effect of foreclosed properties, *Journal of Urban Economics* 66, 164 – 178.

Hornik, Kurt, 1991, Approximation capabilities of multilayer feedforward networks, *Neural Networks* 4, 251 – 257.

Hornik, Kurt, MAxwell Stinchcombe, and Halbert White, 1989, Multilayer feedforward networks are universal approximators, *Neural Networks* 2, 359 – 366.

Lin, Zhenguo, Eric Rosenblatt, and Vincent W. Yao, 2009, Spillover Effects of Foreclosures on Neighborhood Property Values, *The Journal of Real Estate Finance and Economics* 38, 387–407.

Montufar, Guido F, Razvan Pascanu, Kyunghyun Cho, and Yoshua Bengio, 2014, On the Number of Linear Regions of Deep Neural Networks, in Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.: *Advances in Neural Information Processing Systems 27* (Curran Associates, Inc., ).

Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, 2014, Dropout: A Simple Way to Prevent Neural Networks from Overfitting, *Journal of Machine Learning Research* 15, 1929–1958.

Sussmann, Hector J., 1992, Uniqueness of the weights for minimal feedforward nets with a given input-output map, *Neural Networks* 5, 589 – 593.

Titman, Sheridan, Stathis Tompaidis, and Sergey Tsyplakov, 2005, Determinants of Credit Spreads in Commercial Mortgages, *Real Estate Economics* 33, 711–738.

Towe, Charles, and Chad Lawley, 2013, The Contagion Effect of Neighboring Foreclosures, *American Economic Journal: Economic Policy* 5, 313–335.