Repeated measure models for credit scoring

Howard Hamilton, PhD and Jeff Dugger, PhD
27 August 2021

Feature Engineering vs. Feature Learning

Raw credit data

Feature Engineering

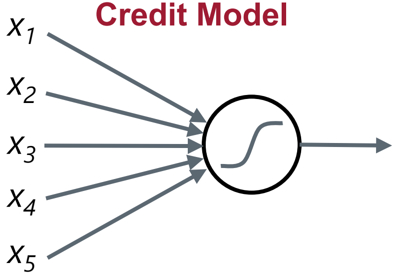
Worst rating trades **current month**

Number of trades worst rating 60 DPD **last 6 months**

Total balance

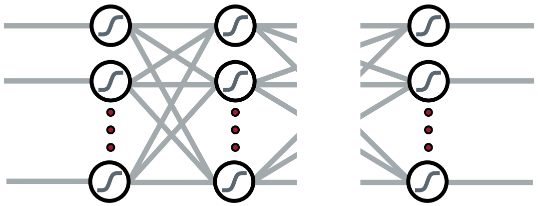
Hand crafted by subject matter experts

Credit Model



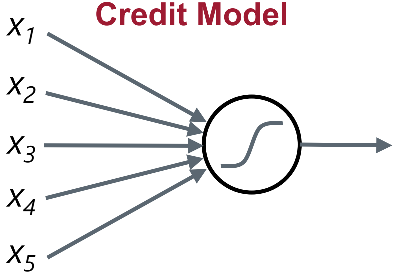
Raw credit data

Feature Learning

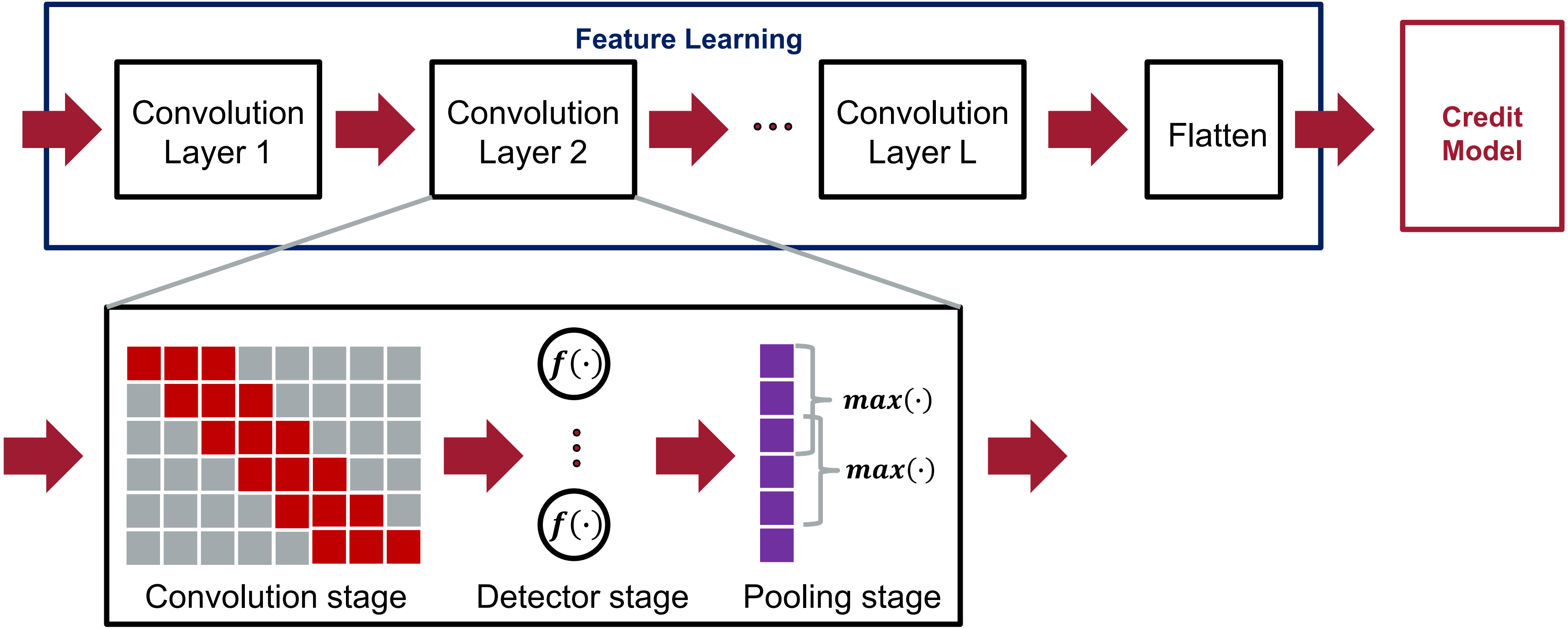


Deep Learning Model

Credit Model

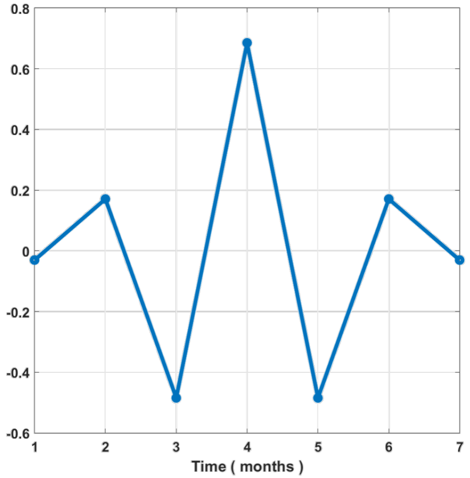


Convolutional Neural Network



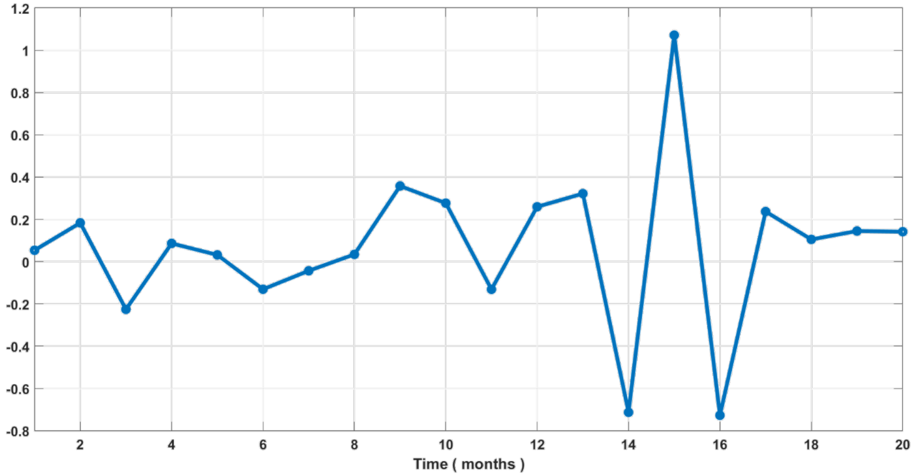
Convolution

Matched filter (desired template)



(vector representation)

Time series containing pattern to be detected

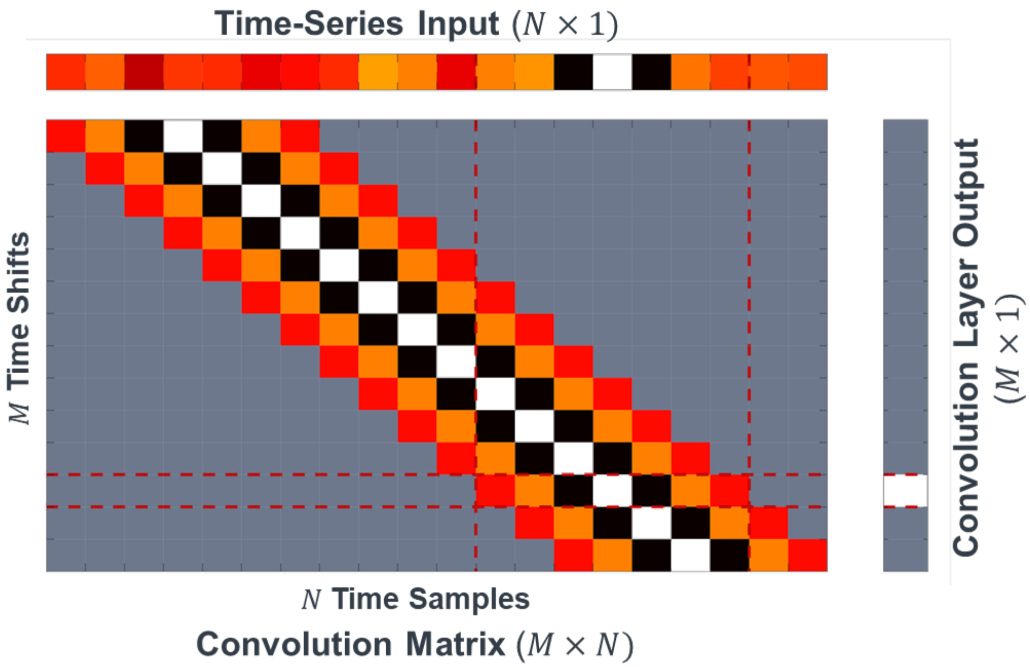


(vector representation)

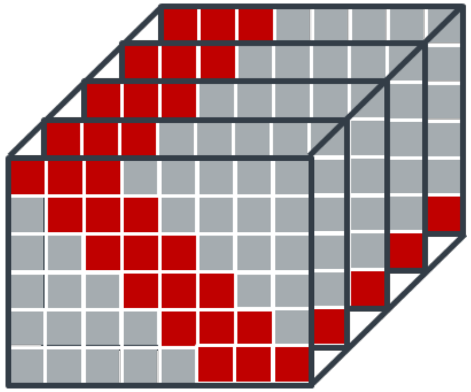
- **Learn template from data**
- **Templates become new features**

- **Convolution as time-reversed cross-correlation**
- **Shifts find the position of patterns in the time series**

Convolution Matrices and "Filter Cubes"



- **Distinct filters match distinct patterns**
- **Represent distinct learned features**
- **Each represented by different matrix**
- **All filters have same length**

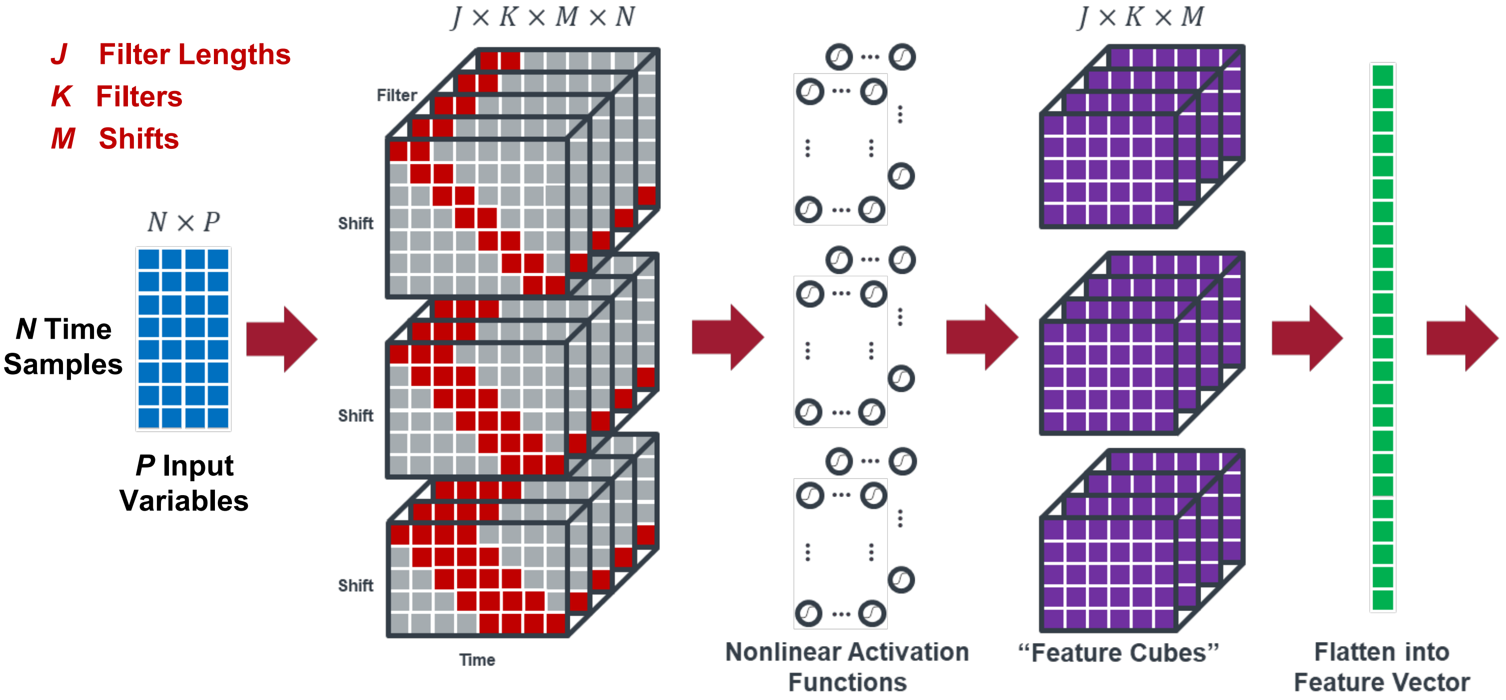


- **Stack matrices into single "filter cube"**

(Tensor—a multidimensional array)

- **Convolution represented as a Toeplitz matrix**
- **Convolution becomes matrix-vector multiply for single filter**

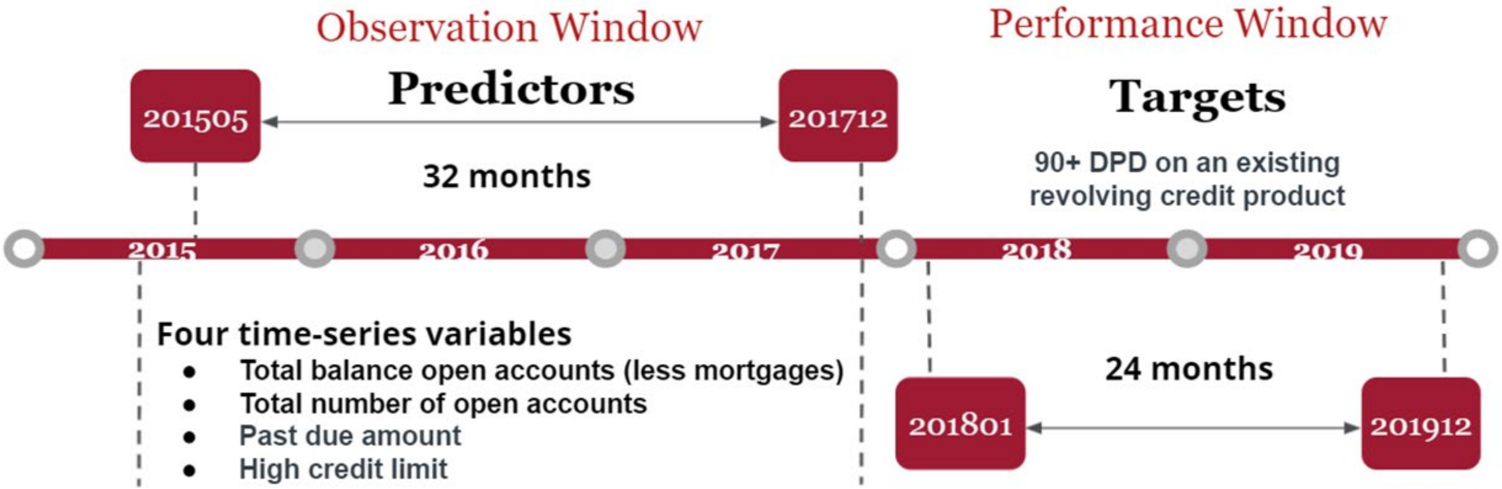
Multiple Inputs, Multiple Filter Lengths



- **Feature vector input to logistic regression model**
- **L1 regularization reduces number of features**
- **Indicates which *filter lengths*, *filters*, and *shifts* matter**

- **Increase number of inputs by increasing columns in input matrix**
- **Increase number of filter lengths considered by stacking "filter cubes"**

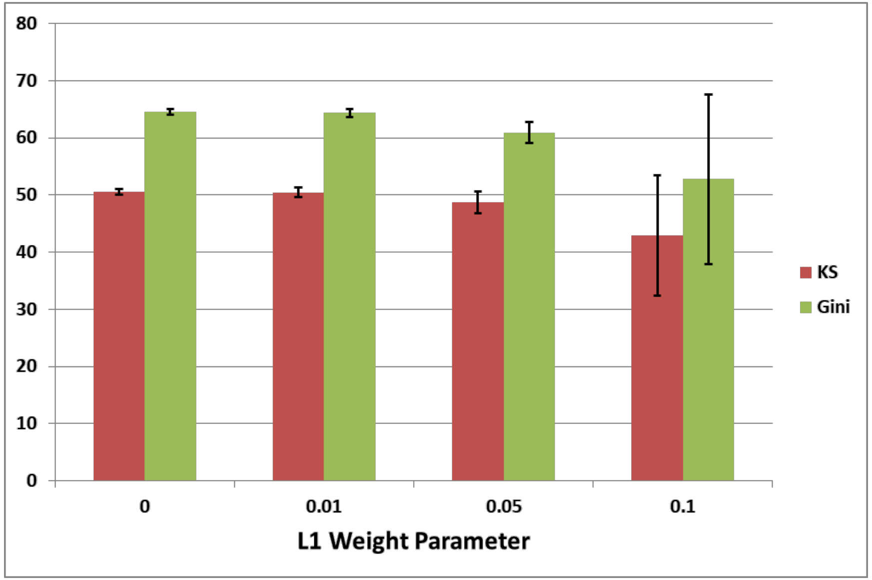
Model Data



- 261,401 consumers
  - 182k development (70%)
  - 78k validation (30%)

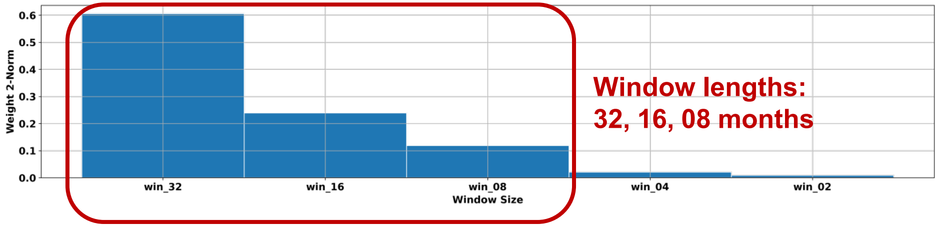
L1 Regularization and Model Performance

- We applied L1 regularization to the logistic regression stage of the TDNN
  - Four different L1 parameters: 0, 0.01, 0.05 0.1
- Performance degrades slightly with increasing L1 parameter, but at the advantage of sharply decreasing number of needed features
- An L1 parameter of 0.01 does not degrade performance by a discernible amount
- For accurate estimates of performance and feature exploration, we ran 30 bootstrap experiments on the data set



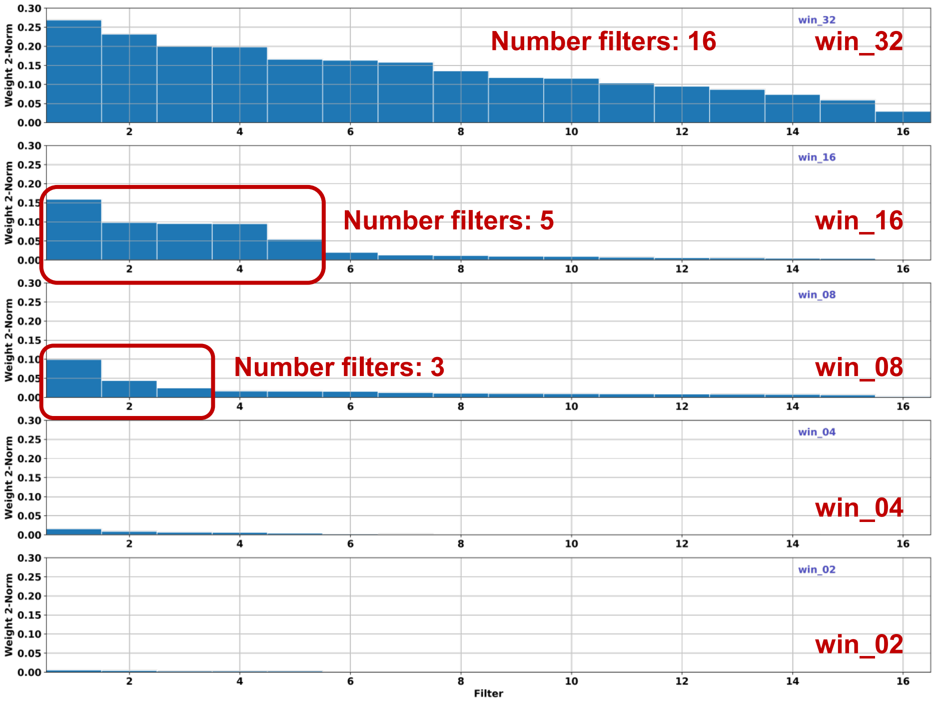
Select Features That Matter

Which window lengths matter?



- **Logistic regression layer weights correspond to *window lengths*, *number of filters*, and *shifts***
- **Apply L1 regularization to logistic regression layer**
- **Averaged weights over 30 bootstrap runs**
- **Utilize Euclidean norm over**
  - **filters and shifts for each window length**
  - **window length and shifts for each filter**

How many filters per window length matter?



EQUIFAX

Conclusions

- **L1 regularization plus judicious choice of resulting weights allowed us to reduce the number of features considered from 1648 to 176**

- **Performance of the model on the reduced feature set did not reduce model performance**

- **Future work will investigate reduction in time-shifts to reduce features further**

- ***Focusing on a handful of features will allow us to investigate the results of the convolutional level in more detail to better understand what the model is actually learning***