

Model beats Wall Street analysts in forecasting business financials

December 19 2019, by Rob Matheson



Credit: CC0 Public Domain

Knowing a company's true sales can help determine its value. Investors, for instance, often employ financial analysts to predict a company's upcoming earnings using various public data, computational tools, and

their own intuition. Now MIT researchers have developed an automated model that significantly outperforms humans in predicting business sales using very limited, "noisy" data.

In finance, there's growing interest in using imprecise but frequently generated consumer data—called "alternative data"—to help predict a company's earnings for trading and investment purposes. Alternative data can comprise credit card purchases, location data from smartphones, or even satellite images showing how many cars are parked in a retailer's lot. Combining alternative data with more traditional but infrequent ground-truth financial data—such as quarterly earnings, press releases, and stock prices—can paint a clearer picture of a company's financial health on even a daily or weekly basis.

But, so far, it's been very difficult to get accurate, frequent estimates using alternative data. In a paper published this week in the Proceedings of ACM Sigmetrics Conference, the researchers describe a model for forecasting financials that uses only anonymized weekly credit card transactions and three-month earning reports.

Tasked with predicting [quarterly earnings](#) of more than 30 companies, the model outperformed the combined estimates of expert Wall Street analysts on 57 percent of predictions. Notably, the analysts had access to any available private or public data and other machine-learning models, while the researchers' model used a very small dataset of the two data types.

"Alternative data are these weird, proxy signals to help track the underlying financials of a company," says first author Michael Fleder, a postdoc in the Laboratory for Information and Decision Systems (LIDS). "We asked, 'Can you combine these noisy signals with quarterly numbers to estimate the true financials of a company at high frequencies?' Turns out the answer is yes."

The model could give an edge to investors, traders, or companies looking to frequently compare their sales with competitors. Beyond finance, the model could help social and political scientists, for example, to study aggregated, anonymous data on public behavior. "It'll be useful for anyone who wants to figure out what people are doing," Fleder says.

Joining Fleder on the paper is EECS Professor Devavrat Shah, who is the director of MIT's Statistics and Data Science Center, a member of the Laboratory for Information and Decision Systems, a principal investigator for the MIT Institute for Foundations of Data Science, and an adjunct professor at the Tata Institute of Fundamental Research.

Tackling the "small data" problem

For better or worse, a lot of [consumer data](#) is up for sale. Retailers, for instance, can buy credit card transactions or location data to see how many people are shopping at a competitor. Advertisers can use the data to see how their advertisements are impacting sales. But getting those answers still primarily relies on humans. No machine-learning model has been able to adequately crunch the numbers.

Counterintuitively, the problem is actually lack of data. Each financial input, such as a quarterly report or weekly credit card total, is only one number. Quarterly reports over two years total only eight data points. Credit card data for, say, every week over the same period is only roughly another 100 "noisy" data points, meaning they contain potentially uninterpretable information.

"We have a 'small data' problem," Fleder says. "You only get a tiny slice of what people are spending and you have to extrapolate and infer what's really going on from that fraction of data."

For their work, the researchers obtained consumer credit card

transactions—at typically weekly and biweekly intervals—and quarterly reports for 34 retailers from 2015 to 2018 from a hedge fund. Across all companies, they gathered 306 quarters-worth of data in total.

Computing daily sales is fairly simple in concept. The model assumes a company's daily sales remain similar, only slightly decreasing or increasing from one day to the next. Mathematically, that means sales values for consecutive days are multiplied by some constant value plus some statistical noise value—which captures some of the inherent randomness in a company's sales. Tomorrow's sales, for instance, equal today's sales multiplied by, say, 0.998 or 1.01, plus the estimated number for noise.

If given accurate model parameters for the daily constant and noise level, a standard inference algorithm can calculate that equation to output an accurate forecast of daily sales. But the trick is calculating those parameters.

Untangling the numbers

That's where quarterly reports and probability techniques come in handy. In a simple world, a quarterly report could be divided by, say, 90 days to calculate the daily sales (implying sales are roughly constant day-to-day). In reality, sales vary from day to day. Also, including alternative data to help understand how sales vary over a quarter complicates matters: Apart from being noisy, purchased credit card data always consist of some indeterminate fraction of the total sales. All that makes it very difficult to know how exactly the credit card totals factor into the overall sales estimate.

"That requires a bit of untangling the numbers," Fleder says. "If we observe 1 percent of a company's weekly sales through credit card transactions, how do we know it's 1 percent? And, if the credit card data

is noisy, how do you know how noisy it is? We don't have access to the ground truth for daily or weekly sales totals. But the quarterly aggregates help us reason about those totals."

To do so, the researchers use a variation of the standard inference algorithm, called Kalman filtering or Belief Propagation, which has been used in various technologies from space shuttles to smartphone GPS. Kalman filtering uses data measurements observed over time, containing noise inaccuracies, to generate a probability distribution for unknown variables over a designated timeframe. In the researchers' work, that means estimating the possible sales of a single day.

To train the model, the technique first breaks down quarterly sales into a set number of measured days, say 90—allowing sales to vary day-to-day. Then, it matches the observed, noisy credit card data to unknown daily sales. Using the quarterly numbers and some extrapolation, it estimates the fraction of total sales the credit card data likely represents. Then, it calculates each day's fraction of observed sales, noise level, and an error estimate for how well it made its predictions.

The inference algorithm plugs all those values into the formula to predict daily sales totals. Then, it can sum those totals to get weekly, monthly, or quarterly numbers. Across all 34 companies, the model beat a consensus benchmark—which combines estimates of Wall Street analysts—on 57.2 percent of 306 quarterly predictions.

Next, the researchers are designing the [model](#) to analyze a combination of credit card transactions and other alternative data, such as location information. "This isn't all we can do. This is just a natural starting point," Fleder says.

More information: Michael Fleder et al. Forecasting with Alternative Data, *Proceedings of the ACM on Measurement and Analysis of*

Computing Systems (2019). [DOI: 10.1145/3366694](https://doi.org/10.1145/3366694)

Provided by Massachusetts Institute of Technology

Citation: Model beats Wall Street analysts in forecasting business financials (2019, December 19) retrieved 27 April 2024 from <https://techxplore.com/news/2019-12-wall-street-analysts-business-financials.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.