Economic Forecasting with Statistical, Hybrid, and Deep-Learning Models

Reid, Andrea
Geomatics and Geodesy Engineering
University New Brunswick
Fredericton, New Brunswick
andrea.reid@unb.ca

Chian Leal, Mary
Geomatics and Geodesy Engineering
University New Brunswick
Fredericton, New Brunswick
mary.chian@unb.ca

Avni, Nadav
Geomatics and Geodesy Engineering
University New Brunswick
Fredericton, New Brunswick
navni@unb.ca

***Abstract*—Effective forecasting methods are invaluable tools that enable users to act more wisely in the present as well as make decisions in anticipation of what is likely to take place. Economic forecasting models attempt to make predictions in such a complex domain that accuracy is significantly lower when compared to modelling in simpler domains. Forecasting models face further difficulty with the introduction of crises, events that are improbable and have unforeseen and confounding effects. Though statistical modelling has evolved to produce predictions that are functionally accurate enough to justify their use in certain cases, the recent introduction of machine learning to modelling has been demonstrated to be superior to statistical modelling alone. More recently a hybrid model has been introduced, representing a bridge between statistical models and machine learning models. This paper explores the use of one such Hybrid Model: NeuralProphet, a descendant of Prophet, a statistical model. Against the backdrop of the COVID-19 crisis, NeuralProphet and Prophet, are used to perform economic forecasting in order to determine their utility as predictive models for economic forecasts. In addition, the efficiency of running multiple models simultaneously is tested by using Pandas, Spark and Databricks.**

Keywords—forecasting, COVID-19, Machine-Learning, time-series, employment, spending, Auto-Regression, NeuralProphet, Prophet, Pandas, Spark, Databricks

I. Introduction

Literature Review. The future is unknown, and "planning of all human activity directed to the satisfaction of needs" is "dependent principally upon correct foresight" and "upon correct advance formulation of their requirements" [4].

Lacking clairvoyance, humans have relied on a variety of methods for predicting future trends in order to maximize their chances of making the most appropriate decisions in the present. Traditionally, outside of subjective methods where "wisdom" or "experience" are used to anticipate future trends, statistical models (SM) dominate [7]. There are a variety of approaches used to overcome insufficiencies in statistical forecasting (e.g. combining forecasts, similarity approach, mixed data sampling, as well as others) [1].

One domain of particular importance for accurate modelling is the economy. Not only is the economy unimaginably complex and overdetermined but predicting human choice with certainty (especially *en masse*) is not within the capabilities of natural sciences [5].

To add to these challenges inherent in economic modelling (EM), the economy is also extremely sensitive to crises, where the reliability of forecasts decreases significantly [1].

Despite these limitations, modelers nonetheless attempt to advance the degree of predictive accuracy present in established models. 'Accurate forecasts require an approach complex enough to incorporate relevant economic data but focused enough to exclude irrelevant data' [2]. The modeler then faces a choice of what data to include and exclude, as well as defining what quantitative and qualitative effect particular data has on the metrics and indices of interest in the model [2]. Though far from perfect, these SMs have a level of accuracy that is considered functionally sufficient for their uses [7]. Despite this, the spirit of human progress demands unceasing advancement in all domains, including in the realm of data science.

Machine-Learning (ML) models, properly tuned and provided with data of sufficient quality and quantity, can demonstrate superior accuracy when compared to traditional SMs. These models automate difficult (and thus error-prone) decisions made by modelers regarding data inclusion/exclusion and the magnitude of effect. This is especially relevant for time-series data, which is prevalent in economic data. Despite the apparent predictive quality of ML models, their complexity generates a barrier for entry for novice users who are more familiar with the traditional models established in their field [2]. Users of ML models must necessarily develop familiarity with the use of ML technologies. Moreover, ML models cannot be used within a field without some expertise in that field. This represents a two-sided barrier for the data scientist and the field practitioner.

Hybrid Models (HM) enable forecasters to harness the predictive power of ML models from without abandoning their proficiency and familiarity with the SM framework. HMs represent a bridge "between classical time series modelling and ML-based methods" [7].

Comparing the Models. This paper performs economic forecasting using a SM (Prophet) as well as ML and HM models (NeuralProphet). Tools used here in the implementation of these models include Pandas, Spark, and Databricks.

II. Data Description and Access

All data was accessed from the Opportunity Insights Economic Tracker GitHub account via direct link using PostgreSQL [8]. The tables are as follows:

- Affinity-State-Daily.csv
- COVID-State-Daily.csv
- Policy Milestones-State.csv
- Employment-State-Daily.csv
- GeoIDs-State.csv

III. Database Management System

We used the library "Pandas" and the Python driver "psycopg2" to have access to the PostgreSQL database (Appendix 1). The process of cleaning data included to remove the empty values and the reformat them in order to optimize the information to be stored into the database.

Furthermore, the code can identify new information from the original source as well as determine if said information should be inserted and the database updated.

Figure 1 shows the total of data that was inserted into the database, as well the number of "null" or "empty" values.

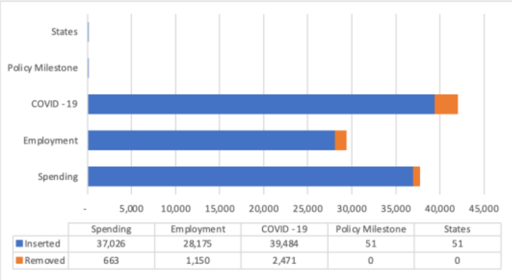


Fig. 1. *Data stored into the database.*

IV. Data Exploration

The data included in the database includes 6 tables. These are as follows:

- A COVID information table that includes items such as 'case count', 'death count', 'test count', as well as other information relevant to the COVID pandemic. This information is in time-series and is broken down by State. A trend to note is that spending and employment dropped with the rise in cases March/April 2020.
- An employment information table that includes employment levels based on income. The data is not absolute but is based on a benchmark from January 2020. The information is in time-series and is broken down by State. A trend to note is that the employment dropped significantly in line with the drop in spending, indicating correlation.
- A Spending table that breaks down spending based on category. The data is not absolute but is based on a benchmark from January 2020. The spending includes consumer spending and does not include government spending. The information is in time-series and is broken down by state. A trend to note is that the spending dropped significantly in line with the drop in employment, indicating correlation.
- A State table that includes the name of the state in short and long form, and the population of the state.
- A policy milestone table, where the columns include the policy implemented, the associated date, the State implementing the policy, and descriptive notes associated with the policy.

For the sake of brevity some of the many possible visualizations are shown below to give the reader an indication of the patterns apparent in the data.

In addition to the correspondence in trend between the variables presented here (Fig. 2, 3, and 4), it should also be noted that our COVID-19 case count data source offers daily updates until April 2022, whereas Consumer Spending is only updated weekly starting in 2022. We thus lack the desired resolution (i.e., daily) for modeling Consumer Spending over the last couple of months. To test our ability to predict economic trends during the pandemic, we can thus use COVID-19 case count, which offers more recent daily data, to help predict Consumer Spending (i.e., our economic target variable).

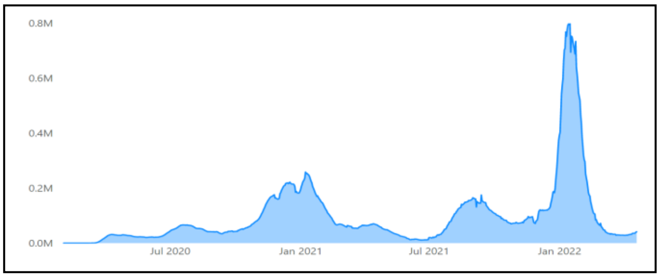


Fig. 2. *US COVID new daily cases from 02 February 2020 to 19 April 2022.*

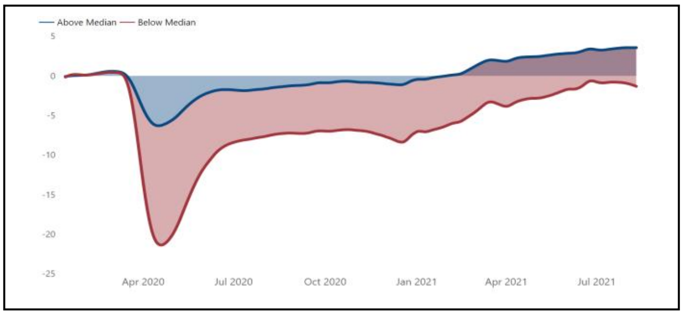


Fig. 3. *US Employment level for above and below median income earners from 03 January 2020 to 20 March 2022. Based on baseline from 04 – 31 January 2020.*

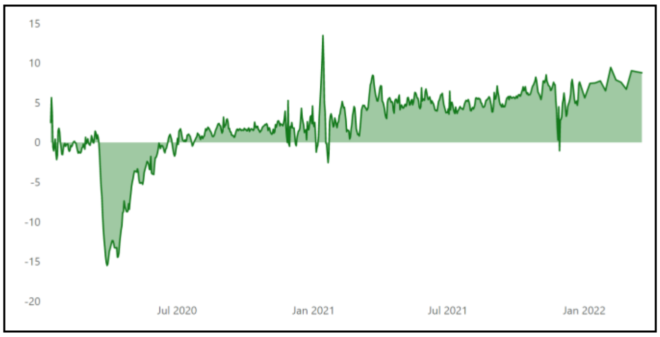


Fig. 4. *Total spending in US Dollars between 13 January 2020 and 20 March 2022. Based on baseline from 04 – 31 January 2020.*

V. Analysis and Prediction

We used Facebook's Prophet [9] and NeuralProphet [10] algorithms to forecast consumer spending. Prophet is an additive regression model that includes a growth function with 'changepoints' to automatically identify points in the data where a change in trend is detected. It additionally uses fourier series to model cyclical patterns in the data that can be attributed to seasonality. NeuralProphet expands on the base structure of Prophet by including terms for lagged and future regressors, as well as an autoregression option in place of trend fitting. It mimics the neural network design of deep learning allowing for a more complex model to solve big data time series problems.

Effect of regressor variable in Prophet. Prophet is used to test the effect of adding a regressor variable to our forecasting models. Using default parameters, including automatic standardization, we compare a univariate model of consumer spending to a multivariate model in which COVID-19 case count is included as the regressor variable using Prophet's 'add_regressor' method.

Figure 5 shows that adding a COVID-19 case count to the analysis changes the prediction, where a drop in spending is forecasted during the COVID-19 wave observed from January to February 2022 (refer to COVID-19 case count figure). This indicates that the addition of a regressor variable may benefit our economic forecasting when data of our target variable (spending) is not yet available.

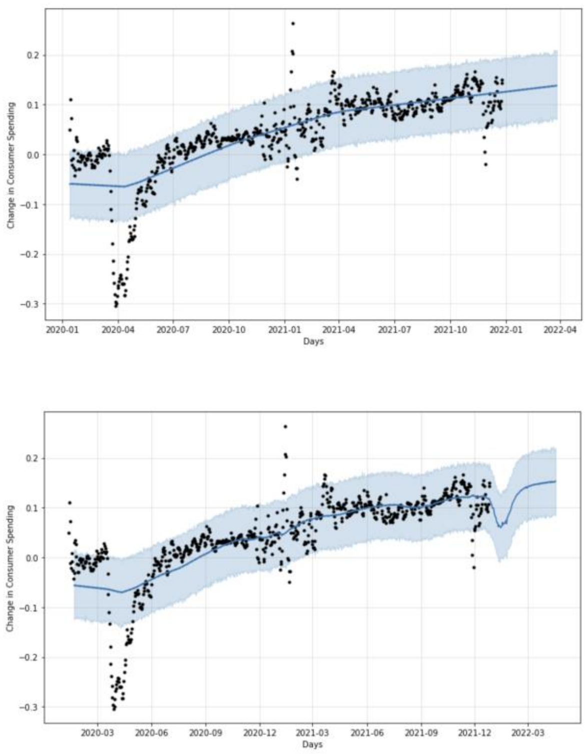


Fig. 5. *Forecasted change in daily consumer spending during the pandemic based on the Prophet model with default settings (top) and with COVID-19 case count as a regressor variable (bottom). Black points are the actual data points, and the blue line is the forecasted spending with 95% confidence level indicated by shaded area.*

Effect of increasing changepoints in Prophet. We also tested the effect of changepoints on the forecasting results by increasing the 'changepoint_prior_scale' from the default value of 0.05 to 10. The 'changepoint_prior_scale' parameter controls the flexibility of the automatic changepoints. A higher number allows for more changepoints to be selected, thus allowing for a better fit to the data.

The results indicate that increasing the changepoint flexibility allows for a better fit of the model to our data points, but at the cost of an increase in uncertainty in our forecasted dates (Figure 6). In order to maintain generality for better forecasting, it is a good idea to optimize the changepoint flexibility and minimize overfitting.

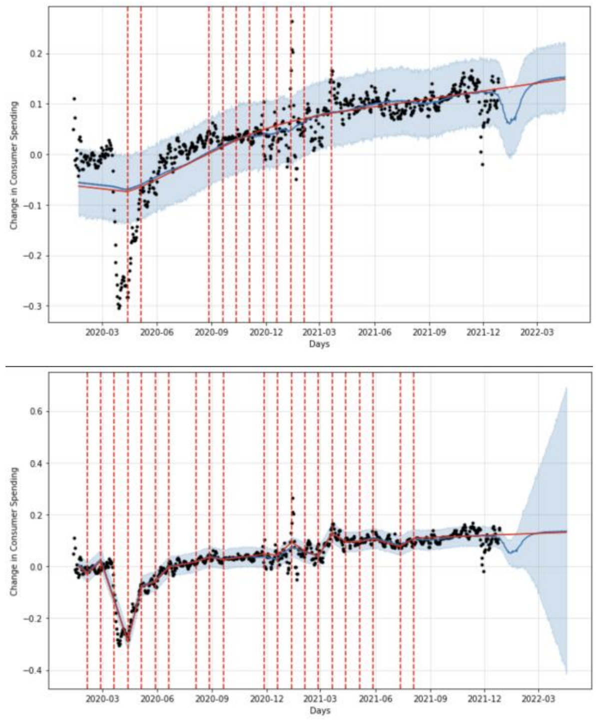


Fig. 6. *Forecasted change in daily consumer spending during the pandemic based on the Prophet model with COVID-19 case count as a predictor variable and 'changepoint_prior_scale' = 0.05 (top) and 'changepoint_prior_scale' = 10 (bottom). Black points are the actual data points, blue line is the forecasted spending with 95% confidence intervals, and trend line is in red with changepoints indicated by vertical red hashed lines.*

Effect of model complexity in NeuralProphet. The effect of neural network modeling on forecasting behavior is tested using four NP models ranging in complexity from a single variable linear regression model to a multivariate autoregression model with a deep neural network (for more details on each model, refer to Table 1).

TABLE I. CHARACTERISTICS OF NEURALPROPHET MODELS.

NP Model	***Model Type***	***Regressor***	***Deep Layers***	***Neurons***
1	Linear Regression	None	0	0
2	Auto-regression	COVID-19	0	0
3	Auto-regression	COVID-19	1	16
4	Auto-regression	COVID-19	4	32

We modeled change in consumer spending using a 1-step ahead (i.e., one-day into the future) approach. For all models, 80% of the data is used for training and 20% for testing. Since we expect consumer spending to change based on daily and weekly patterns, all NP models include daily and weekly seasonality. We use minmax as the standardization method and evaluate performance using the SmoothL1Loss and Root Square Mean Error (RMSE) metrics. For neural network models, the default loss function ('Huber') and optimizer ('AdamW') are used and a time lag of one day for all autoregression models. Batch size and epochs are automatically chosen (recommended for best results).

The results show that the autoregressive models with COVID-19 case count as a regressor perform better than the base model without autoregression (refer to Figure 7). In addition, Figure 7 shows that the two neural network models (i.e., NP Models 3 and 4) have the highest accuracy for training data but at the cost of a lower testing accuracy. This indicates that the deep learning models may be overfitting the data. These results are not unexpected given that our data set is relatively small and deep learning models tend to outperform simpler models only when supplied with 'diverse and complex data' [2].

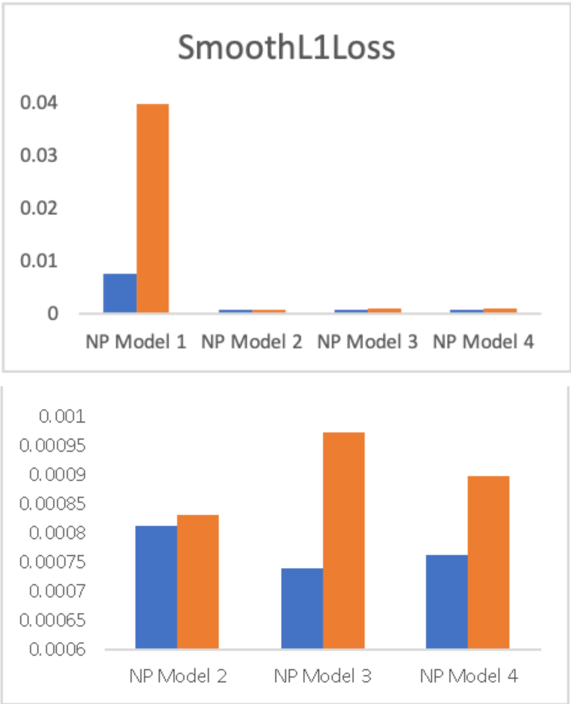


Fig. 7. *Comparison of the SmoothL1Loss metric (epochs = 234) between all four NeuralProphet models (top) and zoomed in to the three autoregression models (bottom), where training data is blue and test data is orange. For details on each model, refer to Table 1.*

Running multiple models with Databricks and Spark. We used Spark and Databricks to run our base model on each of the 52 U.S. States separately. Specifically, we converted our Pandas data frame into a Spark distributed data frame and used pySpark's "PandasUDFType.GROUPED_MAP" function to run our prophet model as a User Defined Function (UDF) grouped by U.S. State. PandasUDF allowed the analyses for each State to be run in parallel. We then compared the processing time to a traditional step-by-step UDF process using Pandas.

We were able to get forecasting results for all States within less than a second using Spark, whereas running the UDF with pandas took 2.61 minutes. Spark's parallel processing and distributed date frame is thus significantly more efficient than running the model step-by-step with Pandas. However, we did notice a potential data loss in our Spark results which requires further investigation into our code.

VI. CONCLUSION

Our results suggest that an autoregression approach to forecasting with a future regressor (i.e., a predictor variable with more current data) is a good candidate for economic modeling during the pandemic. While we did not see an improvement from the deep learning models, larger datasets (i.e., finer resolution time series) may benefit from the added complexity of a neural network. Finally, for running multiple forecasting models simultaneously, Spark is a clear winner over Pandas and offers much potential for real time operational forecasting situations.

Future Work. There is incredible potential for possible work to build on what was explored herein, however only a few recommendations will be made here. Other than running the models again with a change in the parameters, of primary importance is discovering why the data loss occurred with the use of Spark. As well, running the models again with the use of the employment data (collected, cleaned, and added to the database, however not used in the analysis) would add another predictor variable to the forecast. Once the COVID-19 pandemic is 'over', and some semblance of steady-state occurs, then a dataset for the time-period of the crisis can be collected for use in forecasting during a future crisis. Alternatively, the completed dataset can be combined with other datasets from previous crises or geographically disparate crises for creation of an economic meta-dataset consisting of crises from different periods, a variety of locations and arising from various causes.

APPENDIX

Database

REFERENCES

[1] C. Foroni, M. Marcellino, and D. Stevanovic, "Forecasting the covid-19 recession and recovery: lessons from the financial crisis," International journal of forecasting. vol. 38(2), pp. 596-612, 2022. doi:10.1016/j.ijforecast.2020.12.005

[2] A. S. Hall, "Machine learning approaches to macroeconomic forecasting," Federal reserve bank of kansas city, vol. 103(4), pp. 63-82, 2018

[3] R. P. Masini, M. C. Medeiros, and E. F. Mendes, "Machine learning advances for time series forecasting," Journal of economic surveys, July 2021. doi:10.1111/joes.12429

[4] C. Menger, J. Dingwall, and B. F. Hoselitz, "Principles of economics," Libertarian Press, Grove City, PA 1994 (English reprint), original publisher Braumüller, Austria, 1871.

[5] L. V. Mises, "Human action: A treatise on economics," Ludwig von Mises Institute, 1963.

[6] O. Triebe, N. Laptev, and R. Rajagopal, "Ar-net: a simple auto-regressive neural network for time-series," arXiv preprint, *1911.12436*, November 2019.

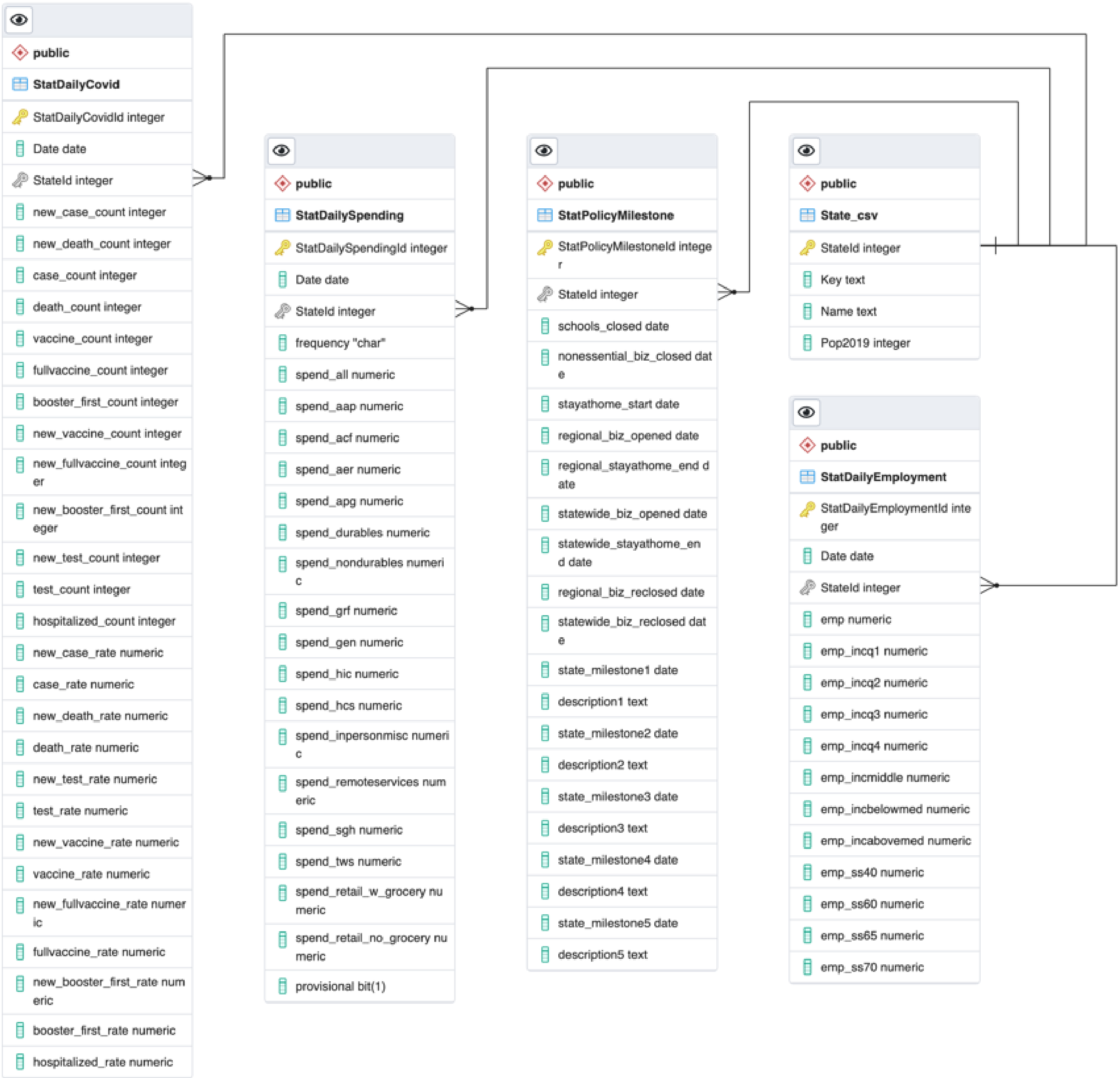
[7] O. Triebe, H. Hewamalage, P. Pilyugina, N. Laptev, C. Bergmeir, and R. Rajagopal, "NeuralProphet: Explainable forecasting at scale," arXiv preprint *2111.15397*, 2021.

[8] https://github.com/OpportunityInsights/EconomicTracker

[9] https://facebook.github.io/prophet/

[10] https://neuralprophet.com/html/contents.html

DATABASE


public
StatDailyCovid
StatDailyCovidId integer
Date date
StateId integer
new_case_count integer
new_death_count integer
case_count integer
death_count integer
vaccine_count integer
fullvaccine_count integer
booster_first_count integer
new_vaccine_count integer
new_fullvaccine_count integer
new_booster_first_count integer
new_test_count integer
test_count integer
hospitalized_count integer
new_case_rate numeric
case_rate numeric
new_death_rate numeric
death_rate numeric
new_test_rate numeric
test_rate numeric
new_vaccine_rate numeric
vaccine_rate numeric
new_fullvaccine_rate numeric
fullvaccine_rate numeric
new_booster_first_rate numeric
booster_first_rate numeric
hospitalized_rate numeric
public
StatDailySpending
StatDailySpendingId integer
Date date
StateId integer
frequency "char"
spend_all numeric
spend_aap numeric
spend_acf numeric
spend_aer numeric
spend_apg numeric
spend_durables numeric
spend_nondurables numeric
spend_grf numeric
spend_gen numeric
spend_hic numeric
spend_hcs numeric
spend_inpersonmisc numeric
spend_remoteservices numeric
spend_sgh numeric
spend_tws numeric
spend_retail_w_grocery numeric
spend_retail_no_grocery numeric
provisional bit(1)
public
StatPolicyMilestone
StatPolicyMilestoneId integer
StateId integer
schools_closed date
nonessential_biz_closed date
stayathome_start date
regional_biz_opened date
regional_stayathome_end date
statewide_biz_opened date
statewide_stayathome_end date
regional_biz_reclosed date
statewide_biz_reclosed date
state_milestone1 date
description1 text
state_milestone2 date
description2 text
state_milestone3 date
description3 text
state_milestone4 date
description4 text
state_milestone5 date
description5 text
public
State_csv
StateId integer
Key text
Name text
Pop2019 integer
public
StatDailyEmployment
StatDailyEmploymentId integer
Date date
StateId integer
emp numeric
emp_incq1 numeric
emp_incq2 numeric
emp_incq3 numeric
emp_incq4 numeric
emp_incmiddle numeric
emp_incbelowmed numeric
emp_incabovemed numeric
emp_ss40 numeric
emp_ss60 numeric
emp_ss65 numeric
emp_ss70 numeric