CASE STUDY: INFLATION FORECASTING WITH ML

Patrick Boily[1,2,3], Andrew Macfie[1], Aditya Maheshwari[1]

Abstract
The Bank of Canada (BoC) uses the Terms-of-Trade Economic Model (ToTEM) as its main projection and policy analysis model; it is used, among other things, to forecast inflation rates. ToTEM is a complex standard dynamic stochastic general equilibrium (DSGE) model, a macroeconomics approach to explain economic phenomena and the effects of economic policy through econometric models based on applied general equilibrium theory and microeconomic principles. In keeping with the society-wide interest to incorporate augmented intelligence methods into traditional situations, BoC is seeking to explore the feasibility of using machine learning to enhance and complement ToTEM's classical formulation and performance.

Keywords
economic forecasting, dynamic stochastic general equilibrium, terms-of-trade economic model, machine learning

Funding Acknowledgement
Parts of this chapter were funded by Idlewyld Analytics and Consulting Services.

[1]Idlewyld Analytics and Consulting Services, Wakefield, Canada
[2]Data Action Lab, Ottawa
[3]Department of Mathematics and Statistics, University of Ottawa, Ottawa
Email: pboily@uottawa.ca

Contents

1. Introduction

The Bank of Canada (BoC) uses the **Terms-of-Trade Economic Model** (ToTEM) as its main projection and policy analysis model. Multiple improvements have been made to ToTEM over the years.

The current version of the model includes multiple interest rates, sector-specific demand specifications for consumption, housing investment and inventory investment, a role for financial wealth in household consumption, and rule-of-thumb price and wage setters.

ToTEM is a standard **dynamic stochastic general equilibrium** (DSGE) model, a macroeconomics approach to explain economic phenomena and the effects of economic policy through econometric models based on applied general equilibrium theory and microeconomic principles.

As a complex DSGE model, ToTEM contains hundreds of non-linear equations; most of its parameters are formally estimated using full information estimation techniques. Linear approximation techniques and econometric methods, combined with a variety of analytical tools, are used to assist forecasting (of inflation, say) and policy simulations.

In keeping with the current appetite for incorporating **augmented intelligence** (AI) approaches to traditional ones, BoC is starting to explore the feasibility of enhancing and complementing (rather than supplanting) ToTEM's classical formulation and performance *via* **machine learning** (ML).

1.1 R&D Avenues

The following tasks were identified as candidates for eventual DV/DS/ML/AI improvements to ToTEM (and related computations):

- significantly **decrease the running time** of the Bayesian DSGE version of ToTEM;
- identify and describe non-traditional (namely, uninformed and **data-driven** ML) approaches to predict inflation rates given historical bank performance and other macro-economic variables;
- set the stage for **hybridized** DSGE/ML approaches to predict inflation, and
- help internal BoC audiences better understand ToTEM-related results *via* **data visualization**.

In this case study, we discuss the second of these.

It should be noted that, as of 3-Sep-2021, **no BoC statement has been made to the effect that ToTEM has been updated to incorporate ML&AI methods**.

1.2 Report Overview

In **Section** 2, we first provide an executive summary of DSGEs (and of ToTEM in particular), followed by a discussion on inflation and how it is measured in a general context, leading to a brief summary of ToTEM's historical performance as a predictive tool to forecast inflation, and a breakdown of DSGEs strengths and weaknesses.

In **Sections** 3 **and** 4, we present some of the advantages of using ML/AI for economic forecasting; suggest a variety of methods (including hybridizations of DSGE and VAR), and describe strategies to obtain a large-enough dataset for ML methods to return benefits. An R notebook illustrating how to implement ML methods for economic forecast can be found in **Section** 5, and a summary of applicable recommendations in **Section** 1.3.

1.3 Recommendations

When it comes to the feasibility study of applying machine learning methods to inflation forecast, the main recommendations are:

1. Given **sufficient data**, machine learning is a powerful tool for unconditional forecasting. Setting up a model for economic data is not trivial, but it is not that difficult either; the crucial point is that **more observations need to be generated/collected** for any ML model to prove useful.

2. We do not suggest displacing existing methods with machine learning, however. Instead it should, in principle, be possible to combine all available models into one **ensemble learning model** that automatically leverages the best qualities of each.

2. DSGEs and ToTEM

In this section, we begin by discussing the Canadian **Terms-of-Trade Economic Model (ToTEM)**, which is a **Dynamic stochastic general equation (DSGE)** model calibrated to Canadian macro-economic data from 1980 until present.

This is followed by some analysis on inflation, specifically in terms of how inflation is caused, how it is currently measured (CPI), and ideas for how to improve this measurement (by using bigger data).

Next, we review the in-sample goodness-of-fit accuracy results from the latest published ToTEM model (known as ToTEM II [15]), as well as the BoC's past forecasting accuracy.

Finally, we present our opinions on the strengths and limitations of DSGE modeling (with a focus on ToTEM II), and introduce the potential avenues for improvement that will be described in more detail through the rest of this report.

Most of the information is taken from [3, 4, 15, 22].

2.1 DSGE Models

DSGE models are used to explain macroeconomic phenomena (behaviour of the economy as a whole; top-down) and to measure the effects of economic policy by using applied general equilibrium theory combined with micro-economic principles (behaviour of individual agents; bottom-up).

Paraphrasing from [26],

> DSGE models study how a complete ["general"] economy's equilibrium supply and demand prices ["equilibrium"] evolve over time ["dynamic"], taking into account the fact that the economy is affected by random shocks ["stochastic"].

As an example, a schematic diagram of a DSGE model from the *Federal Reserve Bank of New York* is shown in Figure 1.[1]

The **Terms-of-Trade Economic Model** (ToTEM) is a DSGE model for the Canadian economy [15]. In this model, the Canadian economy is modelled with four agents:

- **firms**,
- households (**consumers**),
- a **central bank**, and
- a fiscal authority (the **government**).

Firms aim to maximize profits, but face constraints:

- the speed of development of (and access to) production technology;
- labour supply;
- the demand for their outputs, and
- the frequency with which they can modify their prices.

[1]A simple Python-based example can be found here, as well [19].

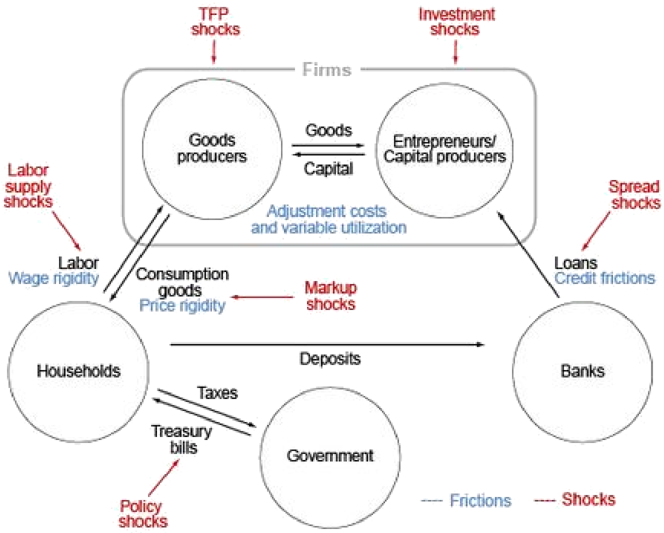


Figure 1. Schematic diagram for the Federal Reserve Bank of New York's 2014 DSGE model [12].

Firms make four types of **finished products**:

- consumption goods and services,
- investment goods,
- non-commodity export goods, and
- government goods.

This production process is divided into two main phases:

1. **intermediate goods** are produced using capital and labour (from households), and commodities and imports (sector-specific), and
2. are then combined into **final goods** and **manufactured inputs**.

The second stage inputs create a sector-specific **fixed-cost of production**; firms try to incorporate a strategic complementarity between price-setting decisions of supplies of different material inputs.

In general, firms face **price stickiness**: they cannot easily modify their prices once set. Consequently, the model price is a function of: the previous price, a weighted mixture of core inflation rate and inflation target, and deviations of markup from the steady state.

Consumers aim to maximize their well-being or utility, subject to budget constraints that limit the rate at which they can accumulate debt.

Consumers sell labour to producers and receive a wage, which is negotiated with the firms; workers re-negotiate their wages "roughly every 6 quarters" [18], which results in a very slow rise in wages over time in the aggregate, regardless of what else is going on in the markets. Consumers are divided into 3 categories:

- **unrestricted lifetime income** consumers can borrow or save to re-allocate consumption over time by trading on short and long-term bond markets, where they get paid a risk and disutility premium to hold long-term bonds;
- **restricted lifetime income** consumers can only trade in long-term bonds, and their consumption decisions are based on long-term rates, and
- **current income** consumers face period-by-period budget constraints which equates their current consumption with disposable income, including government transfers; they cannot access credit or asset markets, so changes in taxes and transfers can have a large impact on consumption.

A country's **Central Bank** aims to maximize the well-being of consumers by minimizing

- deviations of inflation from the target inflation rate;
- output from potential, and
- variability of interest rates (which are competing objectives).

Inflation rises when demand exceeds long-run supply due to marginal cost increases. In ToTEM, firms seek to **maximize profits** in an environment where elasticity of demand for their goods is constant and prices do not fluctuate often (**stickiness**); in this model, inflation is consequently driven exclusively by current and future movements in marginal costs [15].

Central banks can control the overnight interest rate (which influences the **nominal rate over 90-day commercial paper**). This rate does not directly impact spending, however; instead, consumption and investment decisions are based on the entire expected future path of short-term real interest rates (a long-term real interest rate).

Changes in nominal short-term interest rates only influence this long-term real rate because prices and wages are not fully flexible in the short run; monetary policy influences real activity and not just prices, due to nominal rigidity. We cover inflation more thoroughly in Section 2.2.

The **government** is able to place indirect taxes and spend the proceeds of these taxes on consumers according to a set of rules consistent with achieving a pre-specified ratio of debt-to-GDP over the medium term. It can also issue government bonds to domestic and foreign households.

Overall, government budgets are constrained by the

- nominal short- and long-term government bonds,
- level of nominal transfers, and
- sum of consumption and income tax revenues.

One of the primary goals of ToTEM II is to be able to forecast the long-term inflation rate in Canada, especially in the presence of **shocks** (sudden shifts to the production functions of any of the agents described above).

Prior to discussing the known and published accuracy results of ToTEM II, we provide a brief overview of inflation, why its control is important, and some of the drawbacks related to the way it is currently measured, and discuss methods that have been used to track consumer spending patterns more accurately.

2.2 Inflation

Simply stated, **inflation** occurs when the cost of living rises faster than the growth of wages over multiple time periods.

One of the challenges associated with identifying inflation is that measuring the cost of living is **not an objective matter**; not all consumers have the same priorities for expenditures.

In order to standardize the cost of living, different institutions use different measurements, including:

- **Consumer Price Indices** (CPI);
- **Producer Price Indices** (PPI);
- **Gross Domestic Product** (GDP) **Deflators**, and
- **Personal Consumption Expenditure** (PCE).

The BoC uses CPI. Before going into the specifics of how CPI is measured, we first explain the role of central banks and their impact on inflation as a whole.

At a high level, **Central Banks** are responsible for setting policies and regulations [4] relating to:

- employment rates;
- inflation;
- savings;
- credit and banking;
- exchange rate management, and
- market-structuring policies.

The main tool at their disposal to influence all of these factors is **control of interest rates**.

In theory, reducing the interest rates, and consequently making it easier (cheaper) to access credit, promotes an increase in borrowing of money. Higher amounts of borrowing drives higher amounts of spending, and this increase in spending increases the size of the economy.

However, there is also a constraint that exists in the economy surrounding how many goods can be produced; the production of goods relies on factories, machines, materials, worker capacities, etc. In situations where the money available to spend outstrips the growth of production, demand increases enough for goods to become more expensive, resulting in increased inflation.

Inflation does not only happen as a result of increased spending, however; there are in fact various reasons to explain why specific goods become more expensive over time. These can be divided roughly into four categories [22]:

1. as previously discussed, there might be a rise in demand **relative** to supply;
2. **market power** – when a company can avoid competitive pressure (to some extent);
3. **asset markets –** when speculation bids up the price of an asset in the future, causing certain producers to hold onto today's inventory for longer. As a result, short-term supply shrinks, which may cause current prices to increase. When investors who have bet on the future of a product see current prices increasing, they often increase their bets on the future even more, which can lead to more producers holding into their inventory, and so forth;
4. **supply shock** – some natural event (such as pandemics, hurricanes, etc.) may suddenly shrink the ability of a supplier to respond to the demand.

For a company, the benefit of raising prices is clear, but such actions have subtle impacts on the economy as a whole that are worth mentioning.

If a staple good gets more expensive, all consumers generally have to pay more for the product, including people who labour to produce that product. Consequently, labourers in industries where the product price continuously rises should then receive higher wages (in theory, at least).

Thus, if the capital in a system is assumed to stay fixed, the effect of inflation is simply to shift wages from one area into another. If wage increases persist over time, more companies will start producing these products, and the competition will drive the price (and wages) eventually back down.

It is therefore rare, in the present system, for consumers to increase their long-term buying power with inflation – buying power instead gets continuously slightly shifted slightly from one group to another (and back again) [4].

Another point to emphasize is that too little inflation can be as bad (if not worse) in the long run as too much inflation. Inflation is a sign of **economic health**; it indicates that citizens who are working are producing enough value to drive the needs for more goods and services, which can also lead to new products and better supply chains.

While too much inflation can be offset to some extent by restricting access to capital and also by increasing taxes on groups in the economy that are accumulating (and therefore not spending) a high enough proportion of their income, too little inflation suggests, on the other hand, that innovation and progress are **stalling**, which inevitably leads to a decline in demand for goods, as well as a decrease in wages for workers.

In theory, an economy with little inflation should therefore have a higher percentage of unemployed customers, and *vice-versa* [4].

The central bank's goal is thus to balance inflation so that progress can continue to occur, but in such a way that prices do not increase faster than wages. This suggests that in such a system, the supply of available jobs must be slightly smaller than the demand for jobs, to create an "artificial" limit on wage-price inflation.

No "natural unemployment number" is optimal over all time periods [16], but one can estimate what the natural unemployment number that created a perfectly tight labour market would have been at any time in the past.

The only sensible way to approximate this number in **real-time** is to determine if dropping the unemployment rate further actually increased inflation or not [24]. This relationship also states that selecting interest rates (influencing inflation in the process), the central bank is also agreeing to a rate of unemployment that is deemed necessary to hit inflation targets [16].

However, interest rates are not the only factor in influencing interest rates over time. Intuitively, this is understandable; even when borrowing money is cheap, consumers may not opt (or be eager) to borrow. Furthermore, money does not only appears when a consumer adds it to assets portfolio, by borrowing cash or selling securities for cash; inflation does not just grow with demand.

While it is useful to set interest rates in a way that meets with inflation targets historically, it is important to understand the relationship between inflation and other **macro-economic variables** independently from interest rates and other macro-economic variables.

As mentioned previously, the central bank's primary tool for controlling inflation is to set interest rates. The BoC aims to maintain a **target inflation** of 2% per year. But why 2%? Why not 0%? In [17], the authors emphasize that the 2% target accounts for:

- **leaving room to cut rates**, as interest rates tend to be proportional to inflation (rates go up with inflation);
- **avoiding deflation** – since downward price movements, as mentioned above, are potentially harmful;
- **measurement bias** – when CPI (or any other statistic used to measure inflation) shows a 1% or 2% increase, it is likely lower in reality, because the indices don't include all goods and services.

CPI is a **weighted average of the price of a set of goods which resemble the most commonly purchased goods and services in the economy**. The list of goods is updated every two years, *via* a household survey.

There are issues with this statistic – overestimation of the cost of living, for instance, was noted above. Potential reasons include [3]):

1. **substitutions** – consumers buy cheaper substitutes when they are available, and so if a substitute for a commonly purchased good is not on the list within two years it could easily replace an item on the list;
2. **new products** – many popular products penetrate the market in less than the 2-year window;
3. **quality changes** – how many new iPhone versions come out in a 2-year span? What about other technologies? How quickly does technology deprecate?
4. **online retailers** – staple goods can be shipped from global locations (where they may be purchased at a lower price) economically.

The method used to construct the CPI statistic is fraught with issues. **Target survey populations** include families and individuals living in urban and rural private households, but exclude consumers living in colonies, prisons, and chronic-care facilities.

Furthermore, only goods with a **fixed retail price** are included, excluding niche services (such as home renovations, say). Finally, the problems are compounded by the fact that CPI is computed on a monthly basis using only as a sample of the target population, and the contact method may yield an unrepresentative sample.

Our purpose here is not solely to point out the problems with CPI, but also to inform the reader that this statistic has been – and will continue to be – problematic when used to analyze the cost of living.

Specifically, with the large increase in the number of products available to consumer, as well as increasing **socio-economic divides** [29], it might not be reasonable to understand cost of living using solely a **fixed basket of goods** approach; **measures of spending** should also be used.

With the rise of Big Data and the amount of information companies (especially banks) store in a digital format, it would be easy for governments to access total spending data (local, domestic, etc) instead of using CPI surveys. While it could be argued that acquiring this information might constitute an invasion of privacy, this concern could be mitigated if the statistics are provided at an aggregate level rather than an at an individual level.

For instance, in a recently-conducted study at Harvard, researchers analyzed large troves of private credit-card and retail data to understand how American citizens were spending stimulus money, as well as to determine which income and/or demographic groups were the most largely affected by the COVID-19 pandemic [10].

The reason these are relevant to the problem of forecasting CPI is that, in many ways, CPI is already an approximation. If the statistic itself is biased, then even models that perfectly predict CPI still may not actually reflect the condition of society as a whole.

Category	Variables
National Accounts	Consumption, Residential investment, Business investment, Government expenditures, Imports
Prices	Relative price of government goods, Relative price of imports, Core CPI inflation
Labour Market	Labour income tax revenue, Nominal wage
Fiscal Variables	Transfers to persons, Consumption tax revenue, Government debt-to-GDP ratio
Foreign Variables	Rest-of-world output gap, Rest-of-world inflation, U.S. activity measure, Foreign short-term (ST) interest rate, Foreign long-term (LT) interest rate
Interest Rates	Domestic ST and LT interest rate, ST and LT corporate risk premium, ST and LT household risk premium
Commodity Sector	Energy commodity price, Non-energy commodity price
Other	Net foreign assets, Current account-to-GDP ratio, Canada/ROW real effective exchange rate

Table 1. Variables used in ToTEM II (Table 2.1 in [15]).

On the other hand, a measure like **aggregate spending**, which is easier to objectively assess, is also more reasonable to use in monthly comparisons.

For completeness' sake, we cover the definitions of other existing inflation statistics. In the United States, CPI itself has many variants (CPI-U, CPI-W, C-CPI-U), all of which highlight a different group of consumers. The other two primary methods of estimating inflation are:

- **GDP Deflator** – a measure of the change in prices for all goods and services produced in an economy; useful because it helps compare real economic activity from one year to the next, and is not based on a pre-determined basket of goods;
- **Personal Consumption Expenditure** (PCE) – a measure of consumer spending for a period of time; also comes with a PCE Price Index, which estimates whether prices are inflating or deflating with time.

There are also many other variables that can be used to approximate the effectiveness of monetary policy and government spending as a whole, namely: unemployment, wages, taxes, deficit-to-GDP ratio, trade statistics, etc. The full list of variables said to be fed into ToTEM II is shown in Figure 1.

2.3 Performance

Many of ToTEM II's parameters are estimated using the **Covariance Matrix Adaptation Evolution Strategy** (CMA-ES) [15]. **Means** from the quarterly data are used as the benchmarks to calibrate the remaining parameters.

The slope of the empirical **Phillips curve**[2] has also decreased, as has the extent to which exchange rate movements get passed through to the CPI.[3] ToTEM II is designed so that it takes 7 quarters on average for inflation levels to return to the target when the system modelled is faced with typical macro-economic shocks [15]. Because of reduction in persistence of structural inflation, monetary policy does not need to look as far into the future when setting policy, since, all else being equal, maximum impact arrives sooner.

ToTEM II specifically struggles when it comes to estimating the inflation time series because it does not follow a straightforward linear or non-linear trend over time (the trend is **volatile**) and its relationship with the other variables is difficult to define. Still, the prediction error is roughly 0.28 percentage points. More specific details on the accuracy of forecasting for the other estimated variables can be found in the model documentation [15]; the final estimates are shown in Table 2, while the full validation results are shown in Table 3.

Target description	Value
Share of imported goods used in consumption production	0.50
Share of imported goods used in investment production	0.18
Share of imported goods used in exports production	0.33
Ratio of residential structures investment to GDP	0.06
Government expenditures-to-GDP ratio	0.25
Inventory investment-to-GDP ratio	0.004
Commodity production-to-GDP ratio	0.17
Commodity exports-to-GDP ratio	0.14
Investment-to-GDP ratio	0.12
Imports-to-GDP ratio	0.29
Exports-to-GDP ratio	0.31
Relative price of government goods	0.92
Relative price of investment goods	1.26
Relative price of imported goods	1.28
Relative price of exported goods	1.15
Consumption tax revenue-to-GDP ratio	0.18
Labour income tax revenue-to-GDP ratio	0.18
Transfers revenue-to-GDP ratio	0.09
Government debt-to-GDP ratio	0.49
Exchange rate	0.78
Nominal labour income-to-GDP ratio	0.55

Table 2. ToTEM II calibration targets (Table 2.2 in [15]).

Series	Standard deviation of detrended data (in per cent)	Root mean square one-quarter-ahead prediction error (in per cent)	
		ToTEM	ToTEM II
Quarterly core inflation	0.28	0.32	0.28
Nominal exchange rate	6.41	6.54	3.07
Government expenditures	3.59	0.85	0.80
Business investment	9.25	3.11	2.73
Labour hours worked	2.51	0.72	0.57
Imports	8.04	2.59	2.38
Nominal wage	1.95	0.84	0.65
Exports	5.61	2.67	2.23
Output	2.44	0.80	0.58
Consumption+res inv+inventories	4.17	3.48	1.10
Net foreign assets	2.32	2.70	1.13
Domestic short-term interest rate	0.46	0.32	0.32

Table 3. ToTEM II accuracy (Table 2.5 in [15]).

The BoC has retained a history of projections and observed values by date, where at each vintage date the model pro-

[2]The Phillips curve shows the inverse relationship between the unemployment rate and inflation – with economic growth comes inflation, which in turn should lead to more jobs and less unemployment.

[3]Inflation is less sensitive to excess demand and supply pressures, as well as to movements in relative prices such as the exchange rate.

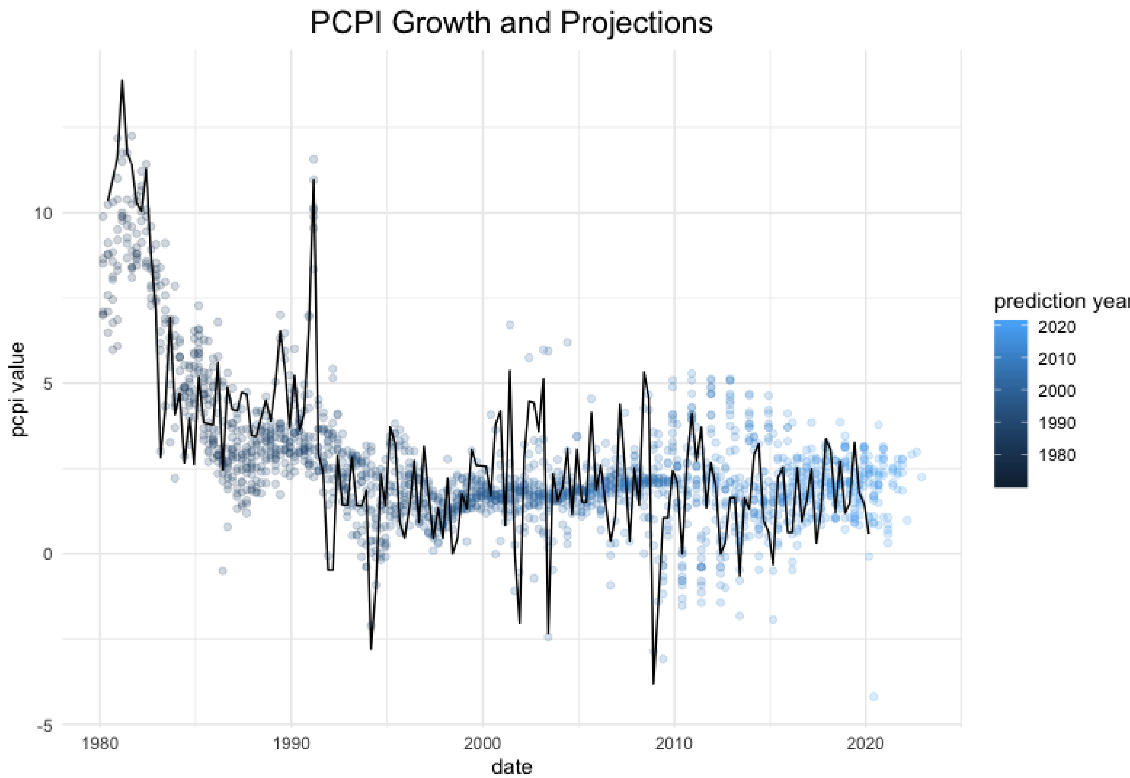


Figure 2. Historical CPI growth values (line) combined with predictions made from various Bank models through the years including ToTEM II (points).

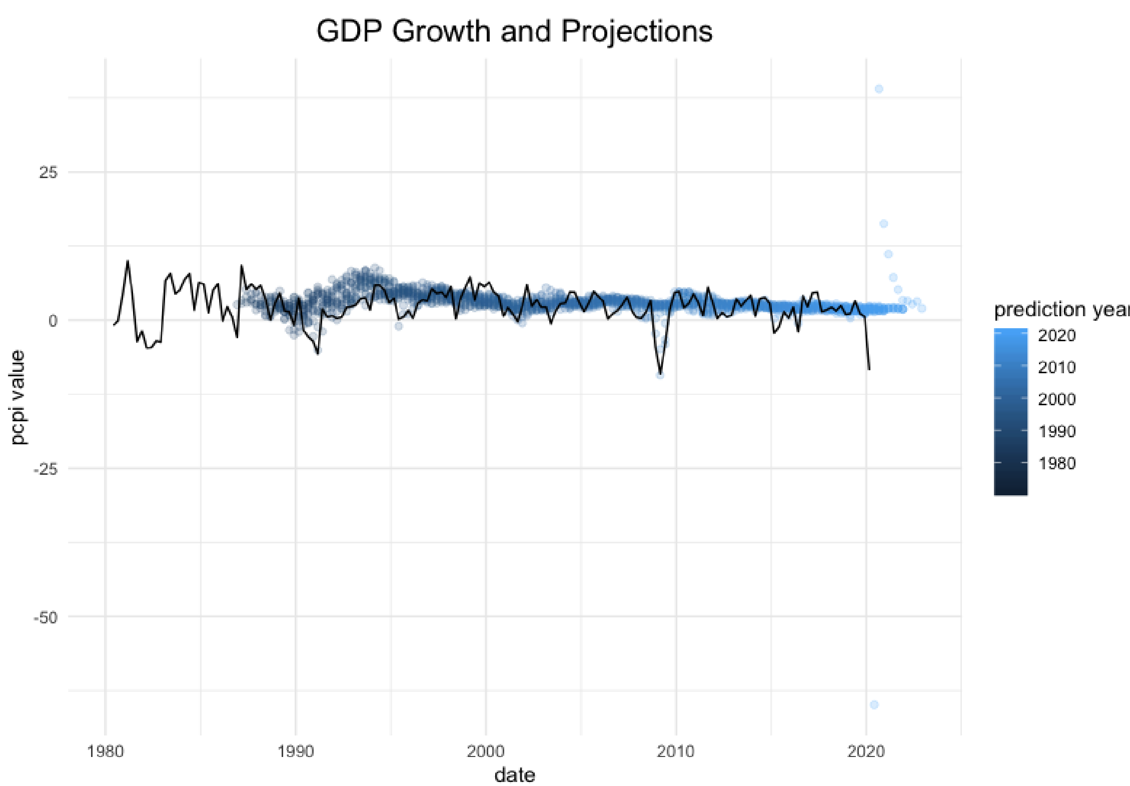


Figure 3. Historical GDP growth values (line) combined with predictions made from various Bank models through the years including ToTEM II (points).

jected roughly 12 quarters ahead. In Figures 2 and 3, we plot the results of their projections for the CPI and GDP growth values, respectively, where the solid line represents the observed values and the points represent the forecasted values.

Multiple projections are made for each quarter, from three years prior up until the quarter actually occurs. Note the outlier values for the GDP predictions in 2020 which provide a false sense of "stability;" absent these outliers, the GDP chart would display much the same level of variability found in the CPI chart.

2.4 Strengths and Limitations of DSGEs/ToTEM

DSGE models guide decisions at many central banks and they remain an active topic in the macroeconomics literature. ToTEM variants have been in use for a number of years at the BoC and have been the subject of significant research and development. They improve over their predecessor, the **Quarterly Projection Model** (QPM) in how they account for **exogenous shocks** and by incorporating structure that make them a **higher-resolution** picture of reality. In the following, we discuss some of the things the ToTEM models do well, and some areas for improvement.

At the most basic level, ToTEM is a powerful and complex DSGE model [32]. It makes extensive but standard economic assumptions which effectively reduce the hypothesis space of relationships between observable variables. This is particularly helpful since there are relatively few Canadian data points available with which to fit the model. The economic knowledge embedded in ToTEM means it would be challenging to achieve similar performance from a less-informed model on a similar dataset; this includes both textbook DSGE models with fewer parameters and **vector auto-regressions**.

DSGE models are **highly interpretable** because there is economic meaning behind each variable and equation. This is in contrast to, say, neural networks where the model is a **black box** that allows computation of outputs from inputs but obscures its internal workings. Interpretability makes it easier to specify priors, perform debugging when aspects of the model fail to correspond to reality, and prove global properties of the model at a glance without computing values across the input space.

Despite all of the above, ToTEM is not quite ideal. Solving for **Arrow-Debreu equilibria** is in general believed to be a computationally intractable task [14]. ToTEM variants linearize equilibrium conditions which introduces some amount of approximation error. It is natural to seek to improve this state of affairs by introducing nonlinear techniques such as those that have appeared in the literature [23]. It may be interesting even with the current approximation scheme to find absolute bounds on or statistical properties of the approximation error, which could be incorporated into overall posteriors [28].

Additionally, computation time is an issue for ToTEM models due to the large number of parameters. The original report contained suggestions on increasing the computational efficiency of Bayesian inference in ToTEM.

As for the **semantics** of the model, any assumptions are always subject to refinement and fine tuning, as exemplified by the evolution from QPM to ToTEM I and from ToTEM I to ToTEM II. There is, presumably, further room for **parametrization**, be it to express bounded rationality or more nuanced strategic behaviour, among others. Ideally new data variables would be found to help fit such new parameters, although with Bayesian priors the model would not be vitiated even with no additional data.

To maximize forecast accuracy, it may pay to **hybridize** ToTEM with an atheoretical model; see Section 3 for a review of the literature on this topic.

DSGEs are generally written in terms of de-trended (**steady-state**) variables. In the current version of ToTEM, de-trended variables are estimated separately from the rest of the model. This may introduce bias; performing joint estimation of the model as a whole would be the statistically principled approach.

Finally, it is natural to ask whether ToTEM could be **simplified**, and parameters removed, while retaining its accuracy. This could benefit both interpretability and computational efficiency.

3. ML and Economic Forecasting

Ultimately, much of economics is concerned with efforts to predict future events. In this section we focus on how two predictive approaches, **DSGE modeling** and **machine learning**, can be combined to produce economic forecasts.

Models are only as good as the **data** that is used to train them. Typically, DSGE models are fit to quarterly time series from the particular economy being studied, and these series are chosen specifically because they have theoretical relationships with the model parameters.

In non-DSGE models (such as machine learning), there is the opportunity and the need to use more data. The **need** exists because the complicated structure of a DSGE is a kind of **information about how the variables interact**, and uninformed/data-driven models need to make up for the lack of this information by using more data.[4]

The **opportunities** for more data come in a few different forms: series from other countries can be used to increase

[4]The fact that atheoretical models do not need to be designed by domain experts is a benefit, all else being equal.

the number of input variables per quarter, as in global vector autoregressions [31], or to increase the number of samples, which is known as **pooling** [5]; the latter is probably more helpful for machine learning.

The set of variables per country can be expanded almost ad infinitum, especially using financial data [9]. However, when adding variables it may be wise to employ basic **regularization** or **priors** (see [25] for the former and the chapter on Bayesian analysis for latter) to give more weight *a priori* to variables deemed most relevant or to encode the fact that some variables probably will not be relevant even if we do not know which ones.

Besides **vector auto-regressions** (VAR) with uninformed priors, **machine learning** deserves attention as a source of atheoretical models. While not as interpretable as statistical methods (see Section 2.4), machine learning is capable of capturing non-linear relationships and achieving state-of-the-art accuracy across a variety of domains.

The recent paper [34] presents a **recurrent neural network** (RNN) for macroeconomic forecasting (e.g. inflation) and favourably compares the results to those of a VAR.[5] Another paper [30] predicted inflation with a random forest model using 122-variable dataset with 672 observations and also found superiority to other methods. A basic machine learning model for predicting inflation is presented in Section 4.

The theoretical ideal in statistics is that the (one) model is in fact the **mechanism of reality** and that we simply do not have full information on the parameter values. If this is not the case and we have multiple useful forecasting models, there are different ways we can use them. Assuming one of the models is the **true model** but we do not know which, the framework of **Bayesian model averaging** can be used to arrive at a posterior distribution over models.

On the other hand, if none of the models are quite right, we can empirically infer which model should be relied on in which context. Model stacking is a machine learning technique to do this automatically, which produces one overall **ensemble model** that combines the output of the component models.

DSGE models may have a role to play as one valuable model among many even if the aim is solely to optimize forecast accuracy and machine learning or VAR were to perform better on that front:

> "In particular, the [DSGE] model compares favourably when forecasting real GDP growth, the trade variables, employment, the real exchange rate, and the short-term nominal interest rate. However, the [DSGE] is less successful when forecasting certain nominal variables, in particular nominal wage growth. [...] [DSGEs] can compete when we use out-of-sample forecast performance as a measure of fit. Naturally, this does not mean that they necessarily 'win' forecasting competitions in all dimensions. [11]"

A number of forecast **hybridizations** have appeared in the literature. Early approaches used a DSGE to set the priors for a Bayesian VAR; competing DSGE models can also be combined to get better performance than any one model in the set [35], as can a DSGE model and a VAR model by **weighing** their forecasts based on the data [2, 20].

In the latter case, the DSGE model receives relatively little weight, although again these models are simplistic compared to ToTEM. It would be natural to consider stacking a DSGE and a machine learning model. Experimentation is key; absent one true model, we are free to try different options and evaluate them with out-of-sample (**testing**) data to determine their merits [6].

Obtaining posteriors rather than point estimates from a machine learning model is possible; **Bayesian neural networks** are popular options, although the likelihood model of an additive noise term is **disappointingly basic** relative to the expressive power of neural nets. The literature on **uncertainty quantification/estimation** in neural networks has alternatives to straightforward Bayesianism[6] [27].

If atheoretical models are found to exhibit greater forecast accuracy than current theoretical models such as large scale DSGEs, we would conclude either that some theoretical assumptions could be improved or that the approximations introduced for computational reasons (e.g. linearization) are detrimental. A machine learning approach may thus be of interest not merely for its own predictive sake but also as a tool for understanding existing methods.

4. Building a ML Model for Inflation

This section proposes a **machine learning** (ML) approach to solving the problem of predicting future inflation (CPI) values given present macro-economic indicators. We also briefly cover how to potentially incorporate historical ToTEM predictions into this modelling framework.

ML models require a **large quantity of observations** from which to make predictions, which in this case are macro-economic indicators observed at different times. We start by discussing the dataset we use in our example of applying ML to predict inflation, including how we expanded the number of observations in the data.

We then elaborate on where more data could be acquired, and identify ML methods that could be applied

[5] The performance of DSGEs relative to VARs has been discussed in the literature [13, Sec. 4.4], but the tested DSGEs tend to be simplistic.

[6] Learning not just $E[Y|X]$ but $Var[Y|X]$ as well.

directly to this data (a more in-depth discussion of ML can be found in the Machine Learning 101 chapter). Finally, we present a **reinforcement learning** (RL) forecasting framework that could also be used to model inflation directly.

4.1 Building a Dataset

ToTEM (and the BoC in general) uses quarterly releases of real-time and historical data with various economic models to produce forecasts for all relevant macroeconomic variables [18]. While the underlying forecasting methods have changed, BoC has stored the historical value of each variable at each date, as well as the future projected values for the next few quarters (including each date).

Because there are about 40 years worth of data, at 4 quarters a year, there are thus only 160 observations for the country as a whole [8]. Furthermore, because the number of **exogenous shocks** that can occur is unknown and their types are diverse,[7] it is necessary to expand the number of observations as much as possible.

To accomplish this, a monthly frequency (which is the most granular option for CPI) can be used, and further, data from additional countries is also used. This dataset has been acquired and built from the Organisation for Economic Co-operation and Development (OECD) data; the specific features include

- **location** (country);
- **time** (year/month);
- **CPI** – with and without food and energy components;
- **GDP** – annual, per hour;
- **interest rates** – short and long term;
- **exchange rates**;
- **employment rate** – part time, temporary, total, self-employment, long-term unemployment, youth, male, female;
- **imports and exports** – goods, services;
- **government** – deficit, revenue, spending, investment in: government, housing, corporate;
- **wages** – labour force, hours per worker, average wages, labour compensation by hour, labour cost per person, labour productivity;
- **population**;
- **PPI**;
- **federal direct investment** – flows in, flows out, stock in, stock out;
- **household** – assets, funds, shares, securities, savings, pension, real wage, income growth, spending, transactions, rent, and
- **material** – consumption and production.

The goal is to combine these variables into a **wide dataset**, where each row is differentiated by the location (country) and time of the observation, and also contains either an observation or an 'NA' (non-applicable) value for that location-time pair. Each of these variables resides in a separate dataset; the wide dataset is merged by outer-joining the datasets together by the location and time fields.

Not all of these features are supplied on a monthly basis however; the outer join leads to a fair amount of 'NA' values in the data. While many ML methods do not breakdown in the presence of such values (decision-tree based methods, for instance), that is not the case for other commonly-used families of algorithms. In Section 5, we present some methods to **impute** the missing values in order to eventually be able to apply Deep Learning models to the data and to generally strengthen the results.

4.2 Data Wish List

This subsection focuses on more potential avenues for improving the data quality used to make predictions. First, it is important to note that outside of the exchange rate, bank rate, and 5-year government bond rate, all variables can (in theory) be reported for smaller regions (and do not have to be recorded at the national level). Second, all non-federal variables should (again, in theory), be able to be computed over any time period (daily, weekly, monthly, etc), especially those related to government revenue and spending, and consumer revenue (wages) and spending.

Collecting the data on a weekly basis,[8] breaking it down by province (even if it is only the larger provinces), and obtaining measurements for different states in the USA, can lead to a larger training set that is similar to Canadian data. This will also solve one issue with using international data; specifically that the definition and importance of many of the macro-economic variables, including Inflation, can change from place to place.

4.3 Problem Breakdown

Most of the variables listed in Section 4.1 can, in theory, be broken down at the provincial or municipal level, except for those federal level variables such as the bank rate, bond rate, and exchange rate. Of course, the BoC can only ever actually control these rates.

Thus, there are two problems that must be solved in order to get an accurate understanding of what impacts CPI:

- determining the impact of societal forces on CPI, and
- determining how different bank rates influence CPI.

Predicting CPI Using Locally Computable Variables The first step is to understand the relationship between variables that are controlled by society at large and the CPI. This is done by using ML to create a complex function that maps historic macro-economic variables to future inflation values.

Either a **recurrent neural network** or an other supervised learning method (if the prediction is always at a fixed interval away) can be used to do this [1, 21].

[7] As we have not experienced all of these shocks in recorded history, we do not actually know how macro-forces would react to them.

[8] Which could also allow for the creation of **momentum variables**.

This can include all variables; but it will be hard to enforce that the federal level variables play an importance proportionate to what is possible in everyday life. Specifically, this model would not be able to enforce that a suddenly increased bank rate would significantly decrease spending and therefore impact inflation, because there have not been enough historical examples of a suddenly increased bank rate. In the case where CPI is accurately predicted without incorporating bank rates, then this model alone would be sufficient to predict future inflation. We demonstrate this modelling framework and assess prediction quality in Section 5.

Learning How Bank Rate Influences CPI There are not enough Canadian examples relating to how the bank rate directly influences CPI; neither are there enough examples of the different shocks that could possibly occur. Therefore, ToTEM II cannot reasonably be expected to provide estimates for non-historical settings that the BoC might nevertheless be interested in investigating. If ToTEM can be sped up to the point where simulation runs take mere minutes (as opposed to weeks), the inputs and outputs of ToTEM can then be fed into an ML algorithm which would learn to output ToTEM computations without having to run such computations, which would then allow BoC to explore the model space, so to speak.[9]

This model (including the bank rates) can be combined with the previous model which predicts CPI using only local-level variables because the ToTEM approximation should receive all the same inputs (as well as some additional federal level variables). In particular, the former could take-in inputs such as "anticipated CPI without ToTEM" and "ToTEM prediction", and use those to predict inflation.

4.4 Other Potential Approaches

From the perspective of the bank, the only real "action" that can be taken to control inflation is to decide on a bank rate (usually increasing or lowering it in incremental units), this problem can be re-casted as a **reinforcement learning** (RL) problem, where an **agent** is rewarded at a higher rate if its inflation prediction is closer to a desired rate and penalized extremely when it makes a problematic decision.

Such an agent can be trained on a series of artificially generated data points.[10] RL already incorporates the idea of maximizing cumulative reward, and has in-built theory to determine how to weigh long term and short term rewards, so assuming the agent has enough data, it will eventually discover a policy that maps the optimal long-term action to each state [21,33].

Furthermore, unlike supervised learning, it is much easier for an agent to react in **real-time** if the action received from the environment is sub-optimal. The challenge with this approach is, once again, the small number of observations; therefore, analysts require robust economic theory to generate additional artificial data.

Because there are many competing theories for which rate to set (for example, on stock trading floors, many fixed-income traders speculate on what the rate decision will be, and similarly, it is likely that different BoC analysts have different preferences), methods that focus on **minimizing regret** given predictions from many "experts" could also prove useful [7].

At any rate, data-driven inflation prediction models require datasets with substantially more observations than those currently used by the BoC.

Overview of Associated Notebook In Section 5, we begin by building the dataset using the sources provided in Section 4.1. We then train a ML model on data up to 1998 and make predictions on all future data-points up to 2020, in order to provide an overview of **out-of-sample testing** (see Machine Learning 101 chapter for an explanation).

Next, using the existing dataset (with missing values), we run a **rolling prediction** supervised learning model which receives all data up to time t minus n, where n represents the **prediction horizon** (in months), and makes a prediction at time t for these n forward periods. We then increment to $t+1$ and repeat the process until we reach the present time. This ensures that the model only trains on past data, and only makes predictions into the future.

We follow this by **imputing the missing values**, either by carrying observations forward, or building mini-models to internally predict missing points. The same rolling prediction framework is applied to the imputed dataset.

Consulting Post-Mortem Feasibility studies are always difficult to navigate, as it is difficult to identify clearly what constitutes a success:

- should the consultants declare the endgame project to not be feasible, the client might agree with the assessment or decide to get a second opinion;
- should the consultants declare it to be feasible, the client might decide to sit on the project, or to go ahead with it, either in collaboration with the consultants or without them.

In three of these outcomes, the consultants might feel that they wasted their time on the project.

Finally, we remind consultants to thread carefully when making suggestions/recommendations to subject matter experts: remain tactful, know your limitations, and remember what you bring to the table.

[9] This can only have a chance of providing useful results if, among other things, many combinations of parameters are tried (including unreasonable ones), and if the results are stored in a dataset containing all input values with corresponding predictions in a separate "vintage" row.

[10] Or real data points from other countries.

5. Direct ML Approach (Notebook)

5.1 Building the Dataset

The following are some libraries and useful functions for building the flat dataset.

```
rm(list=ls())
library(dplyr)
library(ggplot2)
library(cowplot)
library(xgboost)
library(mltools)
library(data.table)
library(Matrix)
library(tidyr)
datadir <- "./FinalData/"
```

```
# Reading data
dataRead <- function(filename, dir = datadir, type = ".csv") {
  read.csv(paste0(datadir, filename, type))
}

# Convert OECD Time Variable, and Keep Only Relevant Features
# Time is converted into a numeric (month) variable, by subtracting the year from 1980
(meaning Jan 1980 is time 0), multiplying the result by 12 (months), and then adding the
month of the observation. An annual observation is given a default month value of 12 (re
presenting the end of the year), and each quarter is mapped to 3, 6, 9, or 12.
cleanRawStats <- function(df, valName) {
  tmp <- df %>%
    transmute(Location = tolower(as.character(LOCATION)),
              Time = tolower(as.character(TIME)),
              #Freq = tolower(as.character(FREQ)),
              Value = Value) %>%
    transmute(Location,
              Time = (as.numeric(substr(Time, 1, 4)) - 1980)*12 +
                if_else(substr(Time, 6, 7) == "", 12,
                        if_else(substr(Time, 6, 7) == "q1", 03,
                                if_else(substr(Time, 6, 7) == "q2", 06,
                                        if_else(substr(Time, 6, 7) == "q3", 09,
                                                if_else(substr(Time, 6, 7) == "q4", 12,
                                                        as.numeric(substr(Time, 6,
7))))))),
              Value)
  colnames(tmp)[3] <- valName
  tmp %>% as_tibble()
}
```

In a flat dataset what we require are one observation per row, and one feature per column. In this case, the features that are common between all Macro-economic variables are the Location, and Time observations. Therefore, we join each csv file, generating one new column for the macro-economic variable observed for each file downloaded.

```
allData <- list.files("./FinalData/", pattern = "*.csv")
allData <- substr(allData, 1, nchar(allData) - 4)

fullDataH <- allData[1] %>% dataRead() %>% cleanRawStats(allData[1])

for (i in allData[2:length(allData)]) {
  fullDataH <- full_join(x=fullDataH, y = i %>% dataRead() %>% cleanRawStats(i), by = c
("Location", "Time"))
}
```

We now filter our data to retain only the Times and Locations

```
# We first sort the features in order of most available data.
fullData <- fullDataH %>% filter(Time >= 0) %>% dplyr::select(Location, Time, cpivalue,
cpinonvolvalue, cpipercent, cpinonvolpercent, irlongterm, irshortterm, sapply(fullDataH,
function(x) sum(is.na(x))) %>% sort %>% names)

# We cannot create cpi stats for countries that do not currently have any measures of CP
I at all, so we must remove those countries from our analysis for now that do not have a
ny "CPI Value" metric.
#Note that doing this also removes australia and new zealand which have CPI data in real
life - just not in this dataset. We will merge that later if necessary.
noCPI <- (fullData %>% group_by(Location) %>% summarise(cv = mean(is.na(cpivalue)), cp =
sum(mean(cpipercent))) %>% filter(cv == 1 & cp == 1))$Location

# We also have to remove all rows that do not contain bank rate data.
noRates <- (fullData %>% filter(!(Location %in% noCPI)) %>% group_by(Location) %>% summa
rise(st = mean(is.na(irshortterm)), lt = mean(is.na(irlongterm))) %>% filter(st == 1 & l
t == 1))$Location

# Outside of China and India, which have no long-term rates and half-completed short-ter
m rates, every other place being removed has no rate data at all. This is fine, we will
go ahead and remove these countries (but keep China and India for now).
fullDataM <- fullData %>% filter(!(Location %in% noCPI), !(Location %in% noRates)) %>% a
rrange(Time, Location)
```

We start by building the output columns, which are future CPI values. In this notebook we stick to predicting 1 month, 1 quarter, and 2 years out.

```
# Create next month, one quarter, and 8 quarter out CPI as output variables
fullDataXG <- fullDataM %>%
  filter(!is.na(cpivalue)) %>%
  mutate(Location = factor(Location, levels = sort(unique(fullDataM$Location)))) %>%
  group_by(Location) %>%
  mutate(cpiVal1m = lead(cpivalue, 1), cpiVal1 = lead(cpivalue, 3), cpiVal8 = lead(cpiva
lue, 24),
         cpiValm1 = lag(cpivalue, 1), cpiValm2 = lag(cpivalue, 2), cpiValm3 = lag(cpival
ue, 3), cpiValm4 = lag(cpivalue, 4), cpiValm8 = lag(cpivalue, 8)) %>%
  ungroup
```

```
# One-hot encode the location factor variable so it is numeric-input friendly
fullDataXG <- bind_cols(fullDataXG,
                        as_tibble(unlist(model.matrix(~ Location, data = fullDataXG)[,-
1]))) %>%
  dplyr::select(cpiVal1m, cpiVal1, cpiVal8, Location, Time, cpipercent, cpivalue, cpiVal
m1, cpiValm2, cpiValm3, cpiValm4, cpiValm8, everything())
```

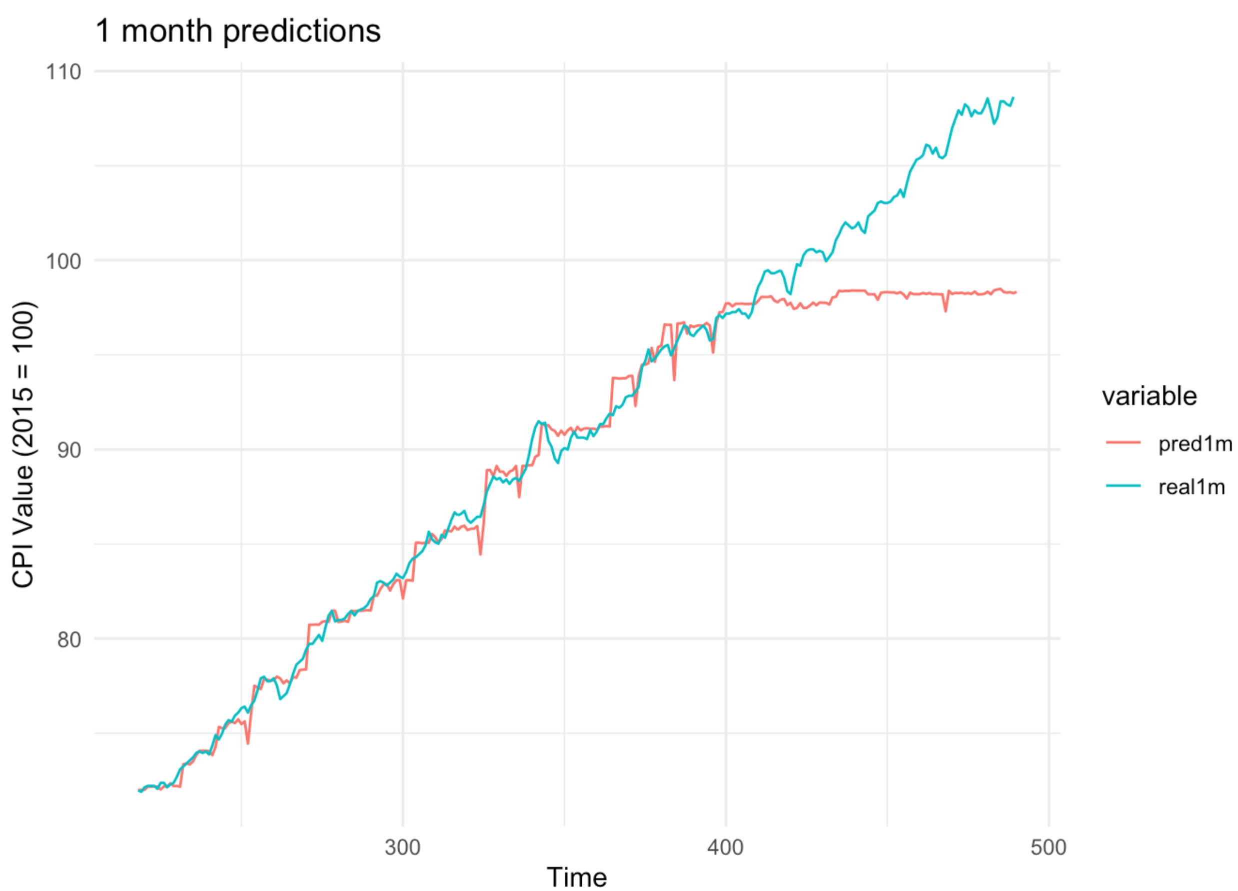
5.2 Clean Train-Test Split Model

Just to get an idea of the "worst-case" scenario for this type of approach, we start with a very basic train-test split where all data before the year 1998 is used to train the model, and all data after is predicted. We also make two sets of predictions, one for the Canada-focused dataset, and one that predicts CPI values for all countries.

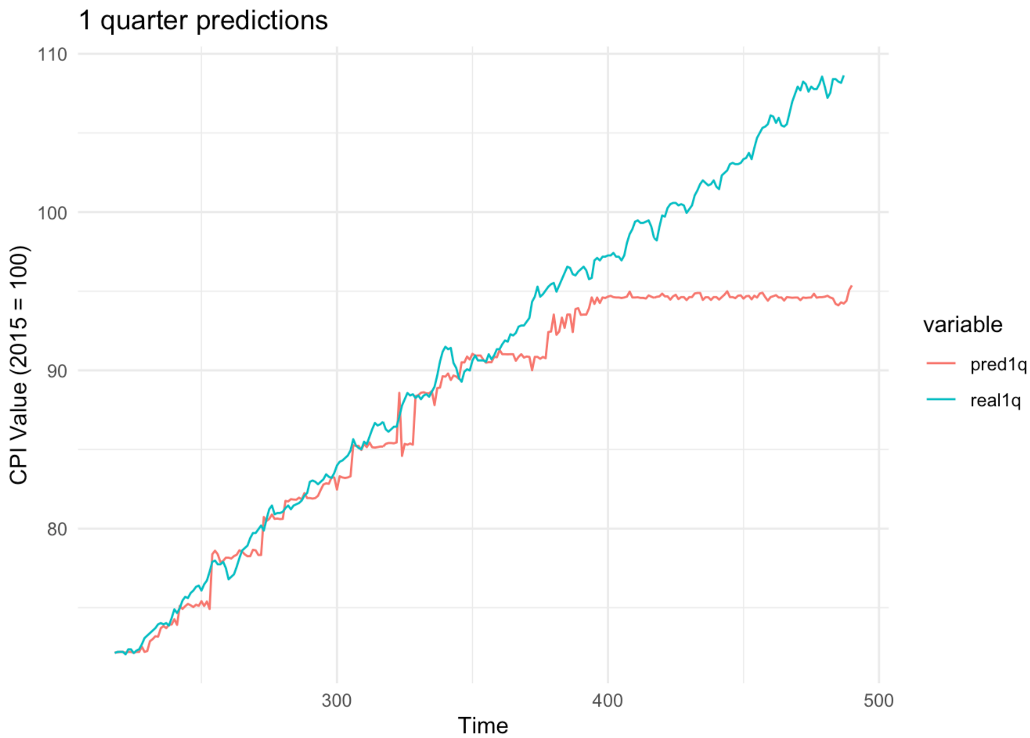
```
trData <- fullDataXG %>%
  filter(Time > 8, Time <= 216) %>%
  dplyr::select(-c(Location, cpiVal1m:cpiVal8)) %>%
  data.matrix

xgbTs = xgb.DMatrix(data = fullDataXG %>% filter(Time > 216) %>% dplyr::select(-c(Locati
on, cpiVal1m:cpiVal8)) %>% data.matrix) #tstLabel1m

trLabel1m <- filter(fullDataXG, Time > 8, Time <= 216)$cpiVal1m
tstLabel1m <- filter(fullDataXG, Time > 216)$cpiVal1m

trLabel1 <- filter(fullDataXG, Time > 8, Time <= 216)$cpiVal1
tstLabel1 <- filter(fullDataXG, Time > 216)$cpiVal1

trLabel8 <- filter(fullDataXG, Time > 8, Time <= 216)$cpiVal8
tstLabel8 <- filter(fullDataXG, Time > 216)$cpiVal8

xgbTr1m = xgb.DMatrix(data = trData, label = trLabel1m)
xgbTr1 = xgb.DMatrix(data = trData, label = trLabel1)
xgbTr8 = xgb.DMatrix(data = trData, label = trLabel8)

xgbf1m <- xgboost(data = xgbTr1m, nrounds=50, verbose=0, objective = "reg:linear")
xgbf1 <- xgboost(data = xgbTr1, nrounds=50, verbose=0, objective = "reg:linear")
xgbf8 <- xgboost(data = xgbTr8, nrounds=50, verbose=0, objective = "reg:linear")

predY1mf <- predict(xgbf1m, xgbTs)
predY1f <- predict(xgbf1, xgbTs)
predY8f <- predict(xgbf8, xgbTs)

tstDataP <- bind_cols(predfm = predY1mf,
                      predf1 = predY1f,
                      predf8 = predY8f,
                      fullDataXG %>%
                        filter(Time > 216) %>%
                        dplyr::select(cpiVal1m:cpiVal8, Location, everything()))
```

As we can see, the model initially predicts the outputs well, but then as time goes on, gets worse at making predictions.

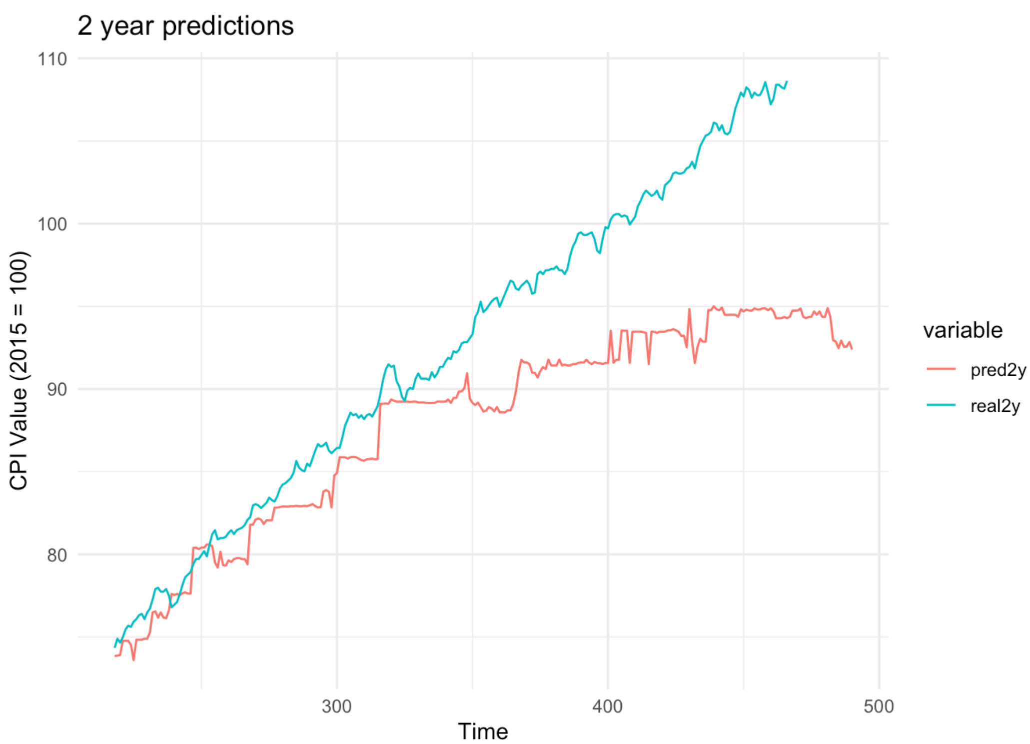
```
# 1 month out predictions and reals
ggplot(tstDataP %>%
         dplyr::select(pred1m = predfm, pred1q = predf1, pred2y = predf8, real1m = cpiVa
l1m, real1q = cpiVal1, real2y = cpiVal8, Location, Time) %>%
         melt(id.vars = c("Location", "Time")) %>%
         filter(Location == "can", Time > 217, variable %in% c("pred1m", "real1m"))) +
  geom_line(aes(x=Time, y=value, col = variable)) + theme_minimal() + labs(x="Time", y="
CPI Value (2015 = 100)", title = "1 month predictions")
```




```
# 1 quarter out predictions and reals
ggplot(tstDataP %>%
         dplyr::select(pred1m = predfm, pred1q = predf1, pred2y = predf8, real1m = cpiVa
l1m, real1q = cpiVal1, real2y = cpiVal8, Location, Time) %>%
         melt(id.vars = c("Location", "Time")) %>%
         filter(Location == "can", Time > 217, variable %in% c("pred1q", "real1q"))) +
  geom_line(aes(x=Time, y=value, col = variable)) + theme_minimal() + labs(x="Time", y="
CPI Value (2015 = 100)", title = "1 quarter predictions")
```




```
# 2 years out predictions and reals
ggplot(tstDataP %>%
         dplyr::select(pred1m = predfm, pred1q = predf1, pred2y = predf8, real1m = cpiVa
l1m, real1q = cpiVal1, real2y = cpiVal8, Location, Time) %>%
         melt(id.vars = c("Location", "Time")) %>%
         filter(Location == "can", Time > 217, variable %in% c("pred2y", "real2y"))) +
  geom_line(aes(x=Time, y=value, col = variable)) + theme_minimal() + labs(x="Time", y="
CPI Value (2015 = 100)", title = "2 year predictions")
```



Of course, in reality, we do not have to make predictions going forward with data before the year 2000 (we can use all the latest known data available). Therefore, we next create some "rolling predictions".

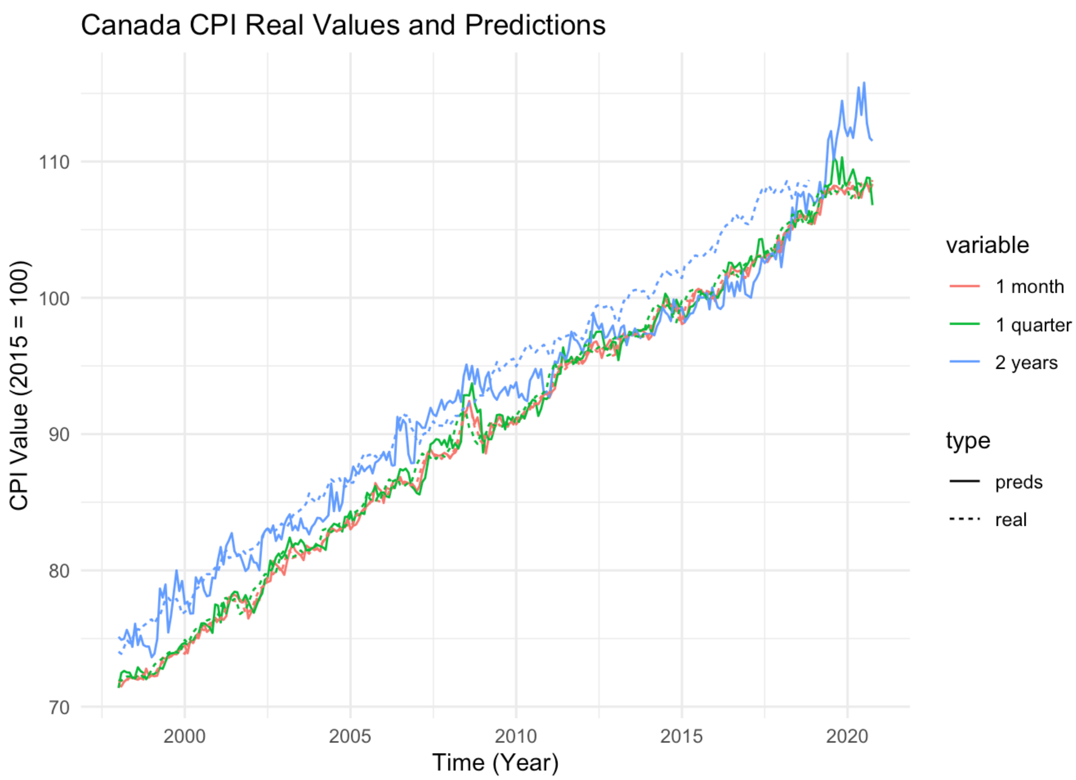
5.3 Rolling Predictions with XGBoost

Starting from 1998, we use all data up until a given time period to predict the CPI value 1 month, 3 months, and 24 months ahead. Then at each new month we build a new model with the latest data.

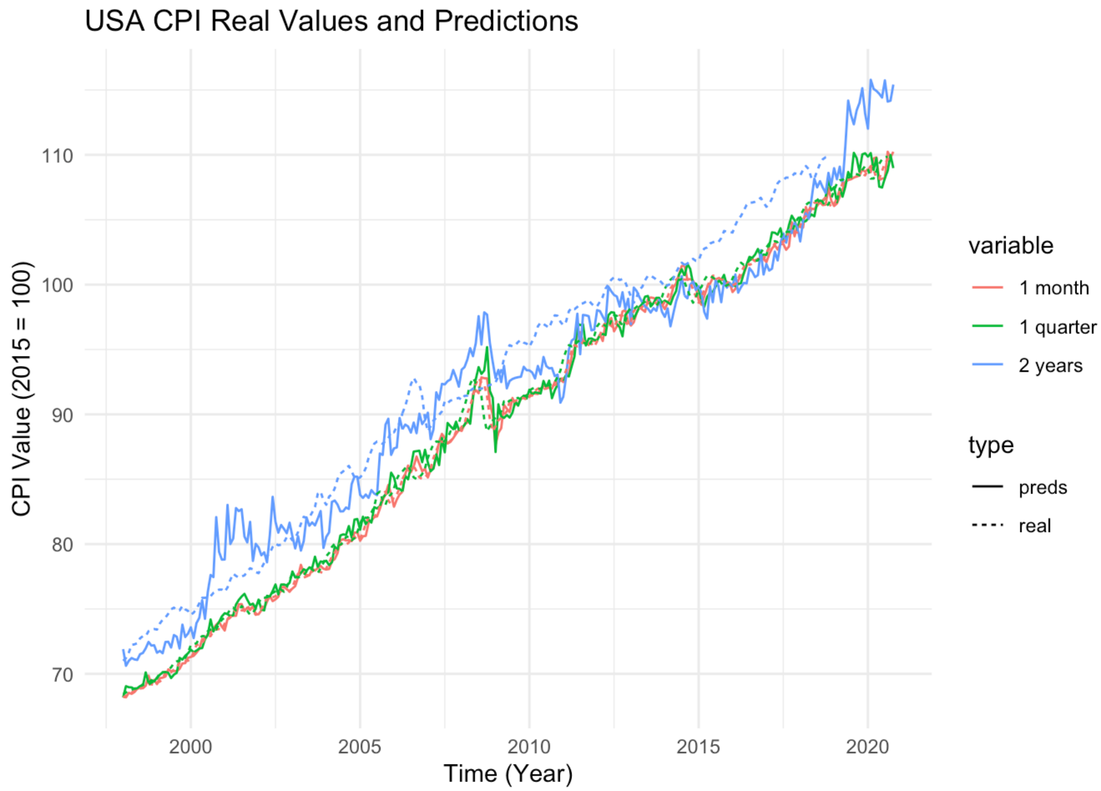
Now we read the constructed dataset, and plot predictions for a few different countries/regions. Results can be summarized as follows:

- Canada, USA, Germany: At the two-year level there is a consistent underprediction at the 2-year out level after 2010. This could be related to the housing crisis. Otherwise, the 1q ahead and 1m ahead predictions are good
- EA19: Really only a dip after 2015 in prediction quality
- India: All predictions consistently underperform
- Japan: Faces much more volatile CPI and inflation conditions, so many predictions are under/over quite frequently.

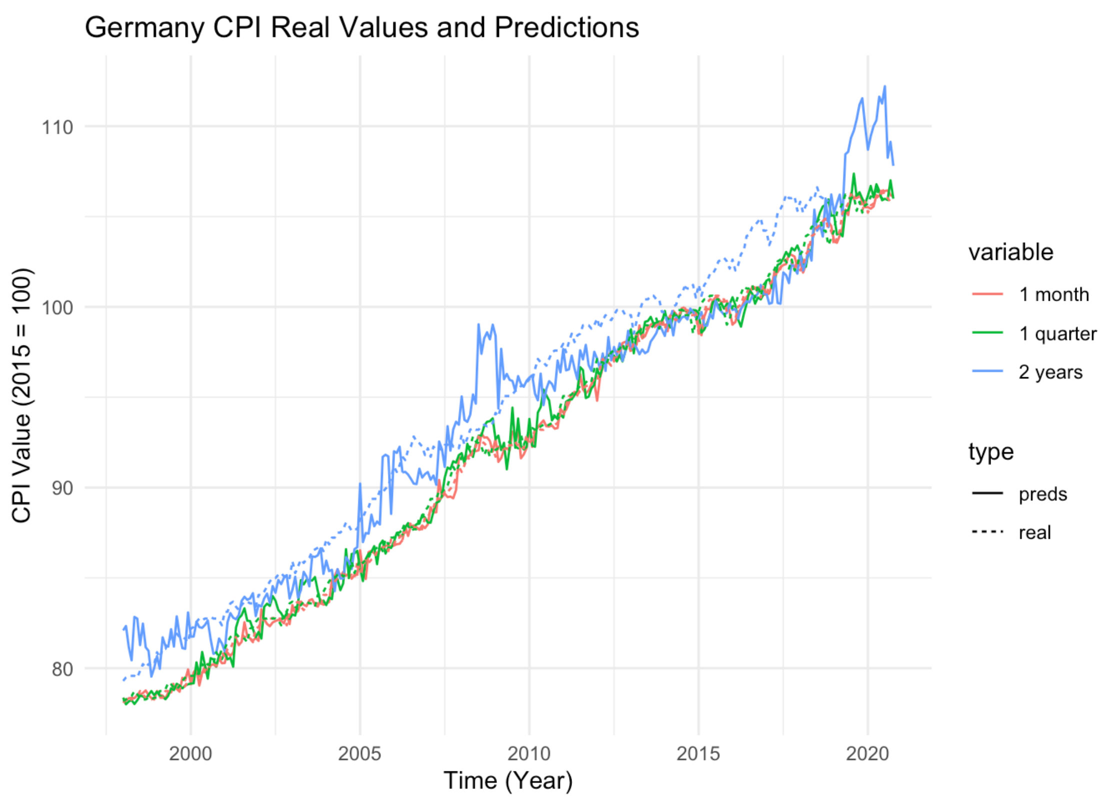
```
predsmm <- readRDS("./XGFullPredsmm.rds")

ggplot(predsmm %>% filter(Location == "can", !is.na(value))) + geom_line(aes(x=1980+Time
/12, y=value, linetype = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value
(2015 = 100)", title = "Canada CPI Real Values and Predictions") + scale_color_discrete
(labels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



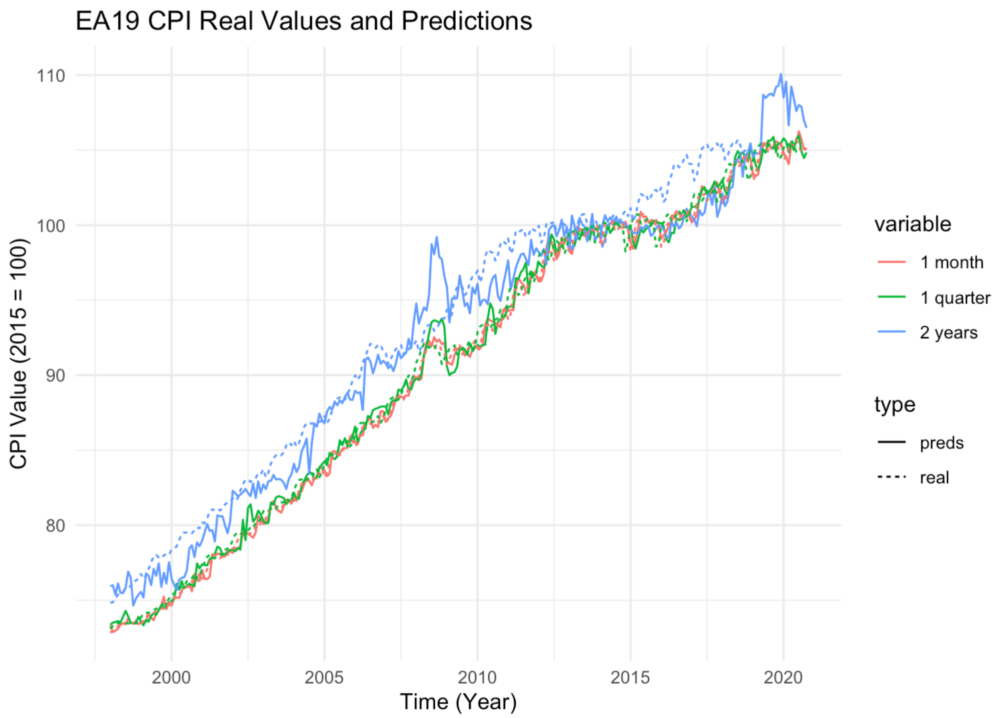

```
ggplot(predsmm %>% filter(Location == "usa", !is.na(value))) + geom_line(aes(x=1980+Time
/12, y=value, linetype = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value
(2015 = 100)", title = "USA CPI Real Values and Predictions") + scale_color_discrete(lab
els = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



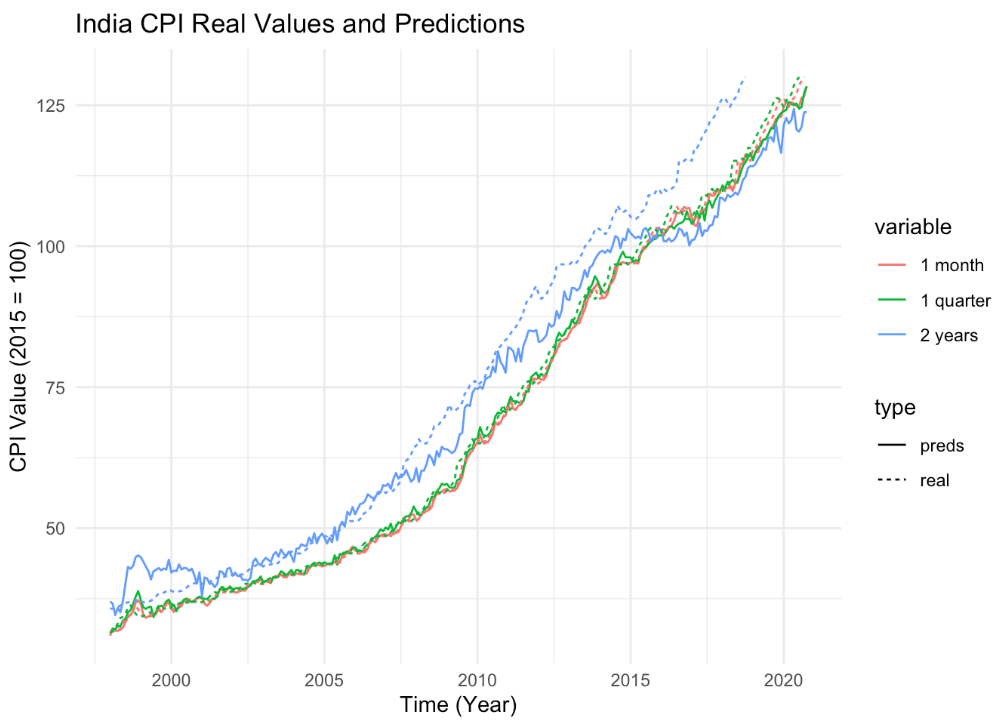

```
ggplot(predsmm %>% filter(Location == "deu", !is.na(value))) + geom_line(aes(x=1980+Time
/12, y=value, linetype = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value
(2015 = 100)", title = "Germany CPI Real Values and Predictions") + scale_color_discrete
(labels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



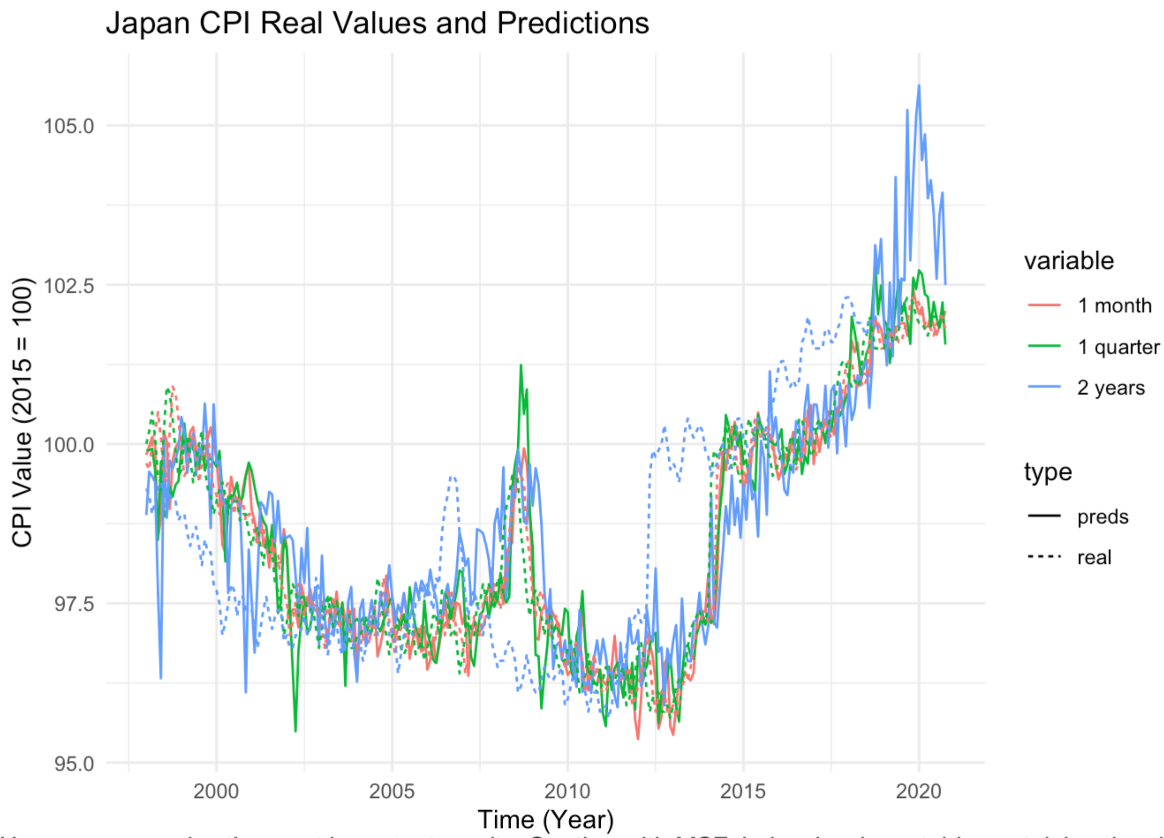
```
ggplot(predsmm %>% filter(Location == "ea19", !is.na(value))) + geom_line(aes(x=1980+Tim
e/12, y=value, linetype = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value
(2015 = 100)", title = "EA19 CPI Real Values and Predictions") + scale_color_discrete(la
bels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```




```
ggplot(predsmm %>% filter(Location == "ind", !is.na(value))) + geom_line(aes(x=1980+Time
/12, y=value, linetype = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value
(2015 = 100)", title = "India CPI Real Values and Predictions") + scale_color_discrete(l
abels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



```
ggplot(predsmm %>% filter(Location == "jpn", !is.na(value))) + geom_line(aes(x=1980+Time
/12, y=value, linetype = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value
(2015 = 100)", title = "Japan CPI Real Values and Predictions") + scale_color_discrete(l
abels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



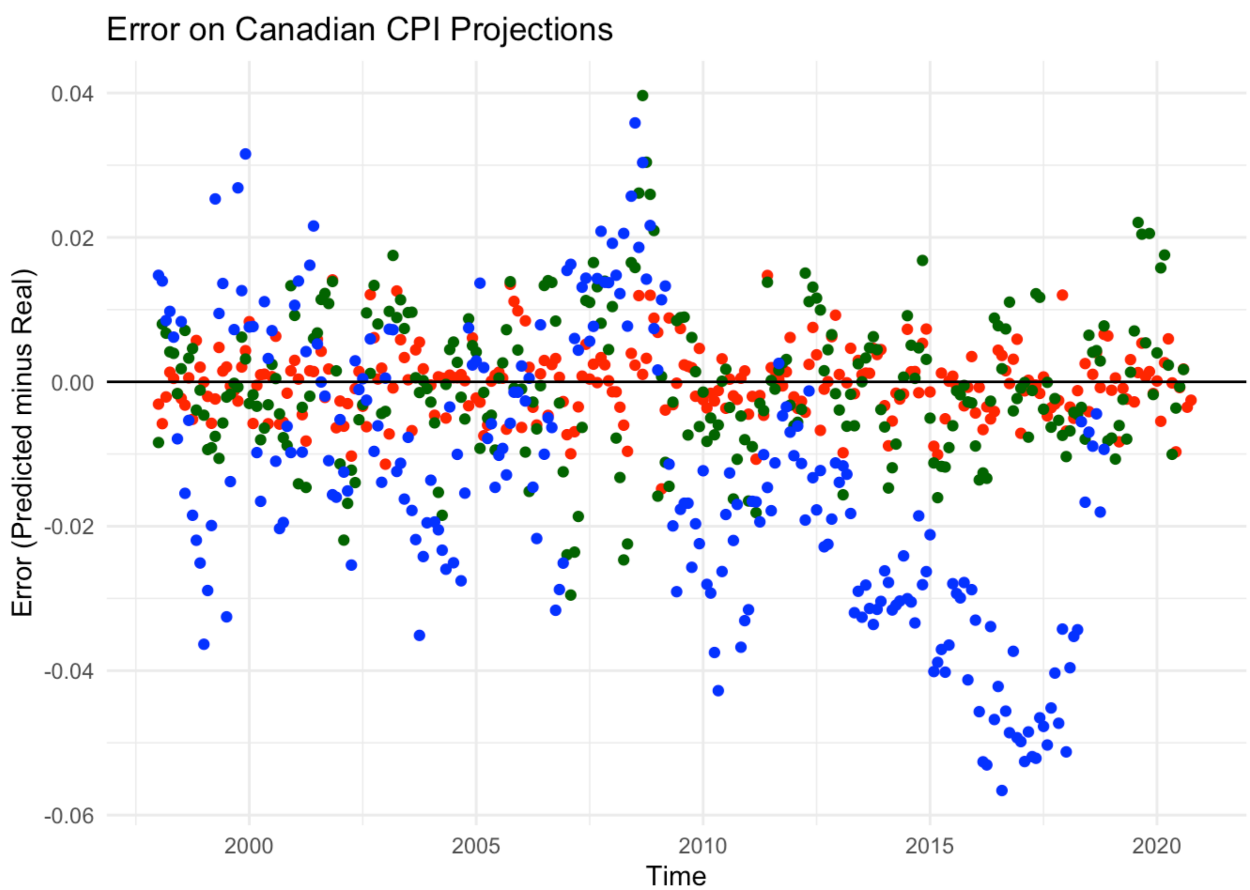
Here we summarise the most important results. Starting with MSE, below is a large table containing the absolute errors and MSe stats for predictions for the whole time series by location.

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 41 x 7
##    Location oneMonth oneMonthMSE oneQuarter oneQuarterMSE twoYear
##    <fct>       <dbl>       <dbl>      <dbl>         <dbl>   <dbl>
##  1 can      -1.56e-2       0.190  -0.0278           0.827  -1.39
##  2 aut       7.27e-3       0.175   0.0131           0.470  -0.938
##  3 bel      -1.89e-2       0.165  -0.0248           0.584  -1.48
##  4 che      -9.49e-2       0.303  -0.148            0.575  -0.885
##  5 chl       8.01e-3       0.225  -0.000709         0.865  -2.28
##  6 chn      -2.75e-2       0.463  -0.113            1.69   -2.25
##  7 col      -1.97e-1       0.495  -0.495            2.09   -3.51
##  8 cri      -2.73e-4       0.159  -0.00653          0.634  -1.84
##  9 cze      -3.92e-2       0.248   0.0245           0.944  -1.12
## 10 deu      -2.24e-2       0.157   0.00935          0.597  -0.923
## # … with 31 more rows, and 1 more variable: twoYearMSE <dbl>
```

Next we plot the errors by percentage below, and we can see that most of the errors are within two percent, except for predictions 2 years out, which generally underpredict after 2010.

```
predsmmW %>%
  filter(Location == "can") %>%
  ggplot +
  geom_point(aes(x=1980+Time/12, y=(m1_preds-m1_real)/m1_real), col = "red") +
  geom_point(aes(x=1980+Time/12, y=(q1_preds-q1_real)/q1_real), col = "darkgreen") +
  geom_point(aes(x=1980+Time/12, y=(q8_preds-q8_real)/q8_real), col = "blue") +
  labs(x = "Time", y = "Error (Predicted minus Real)", title = "Error on Canadian CPI Pr
ojections") +
  geom_hline(aes(yintercept=0)) +
  theme_minimal()
```



Finally, we give our one-quarter-ahead root-mean-squared-errors (in percent).

```
100*(chg %>% summarise(m1mse = sqrt(mean(m1c^2, na.rm=TRUE)), q1mse = sqrt(mean(q1c^2, n
a.rm=TRUE)), q8mse = sqrt(mean(q8c^2, na.rm=TRUE))))
```

```
##       m1mse     q1mse    q8mse
## 1 0.4827698 0.9963039 2.397155
```

5.4 Dealing With Missing Values

While XGBoost is able to handle NA values because it is a tree-based algorithm and can utilize the information that a value is missing as a factor in making splits, regression, deep learning, and many other algorithms that minimize loss functions using matrix operations need actual numeric replacements in order to perform adequately. In the dataset acquired, NA values occur for one of 3 reasons:

1. A variable is collected at an annual or quarterly frequency, and therefore the months in between do not have data.
2. A country did not previously record one of the variables, and so prior to the time it began measuring a value, all other time periods have missing values.
3. The latest data is published at different times for different countries, meaning that in some cases some of the latest observations can be missing.

Our focus in this section is on dealing with the first case; specifically, finding a way to represent annual and quarterly variables in monthly terms. There is no "correct" way to do this imputation for any random dataset; this step requires a series of assumptions driven by domain expertise. We summarize the process of consolidating the variables in list form.

While the OECD data page says a frequency, we will verify the frequency of each variable below, noting that annual data will have 11/12 observations missing, and quarterly data will have 2/3 observations missing. As we can see below, particular variables in a location can be more than 90% and even 100% missing.

```
# Missing data for chosen variables across all countries
fullDataXG %>%
  dplyr::select(Location, Time, cpivalue, irlongterm, irshortterm, gdpannual, uerate, exchRates, population, PPP,
exports, imports, goodsnet, govspendingtotal, shareprice, rent, housenominal, laborutilization) %>%
  group_by(Location) %>%
  summarise_all(function(x) mean(is.na(x)))
```

```
## # A tibble: 41 x 18
##    Location  Time cpivalue irlongterm irshortterm gdpannual uerate
##    <fct>    <dbl>    <dbl>      <dbl>       <dbl>     <dbl>  <dbl>
##  1 aut          0        0      0.246       0.232     0.916 0.322
##  2 bel          0        0      0           0         0.916 0.0774
##  3 can          0        0      0           0         0.916 0
##  4 che          0        0      0           0         0.916 1
##  5 chl          0        0      0.623       0.534     0.931 0.151
##  6 chn          0        0      1           0.354     0.925 1
##  7 col          0        0      0.564       0.149     0.929 0.664
##  8 cri          0        0      0.817       0.800     0.941 1
##  9 cze          0        0      0.313       0.0670    0.919 0.0698
## 10 deu          0        0      0           0         0.916 0.273
## # … with 31 more rows, and 11 more variables: exchRates <dbl>,
## #   population <dbl>, PPP <dbl>, exports <dbl>, imports <dbl>,
## #   goodsnet <dbl>, govspendingtotal <dbl>, shareprice <dbl>, rent <dbl>,
## #   housenominal <dbl>, laborutilization <dbl>
```

```
# Missing data for chosen variables by Time
fullDataXG %>%
  dplyr::select(Location, Time, cpivalue, irlongterm, irshortterm, gdpannual, uerate, exchRates, population, PPP,
exports, imports, goodsnet, govspendingtotal, shareprice, rent, housenominal, laborutilization) %>%
  dplyr::filter(Time >= 360) %>%
  group_by(Time) %>%
  summarise_all(function(x) mean(is.na(x)))
```

Our idea for filling NA values is to simply create linear functions that map the NA value to the average of the nearest two observations in all situations. This type of function is built into the zoo library in R, known as `na.approx`. By setting the maxgap to equal 24, we can enforce that this approach does not start interpolating more than two consecutive years of missing data for any observation. We replace remaining NA values by setting them to the minimum value of each column.

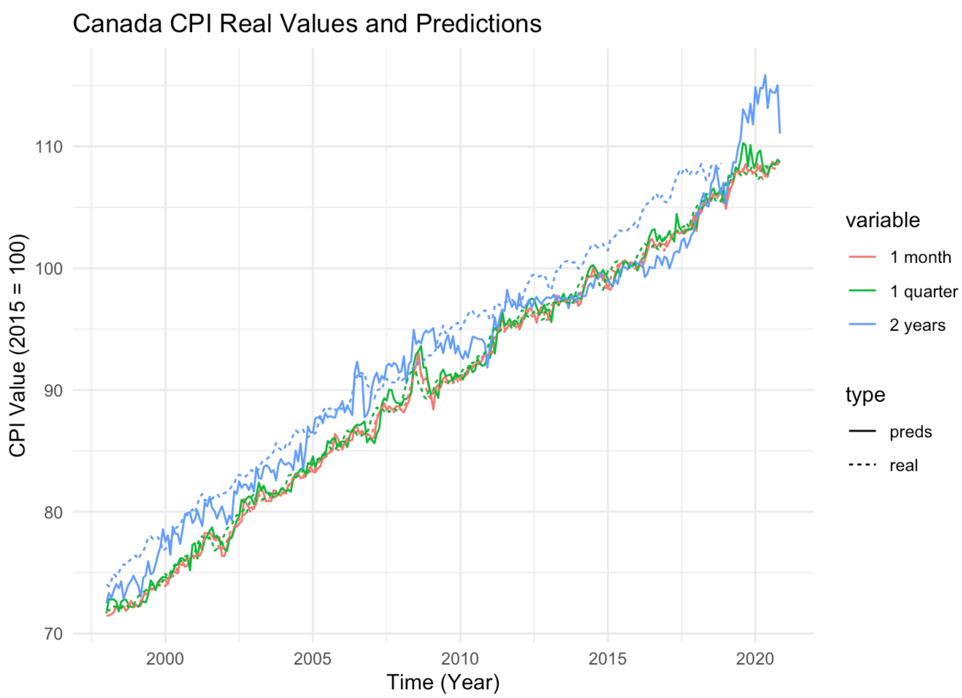
```
# replace na approx
filledDataXG <- fullDataXG %>%
  arrange(Location, Time) %>%
  group_by(Location) %>%
  mutate_at(vars(cpinonvolvalue:govspendingtotal),
            function(x) na.approx(x, na.rm=FALSE, maxgap = 24)) %>%
  ungroup

# fill minimum variable
filledDataXG[,6:131] <- apply(filledDataXG[,6:131], 2, function(x) replace_na(x, min(x, na.rm=TRUE)))

#saveRDS(filledDataXG, "./XGfilledData.rds")
```

```
filledDataXG <- readRDS("./XGfilledData.rds") %>% ungroup
predsFilled <- readRDS("./XGFillPreds.rds") %>% ungroup
```

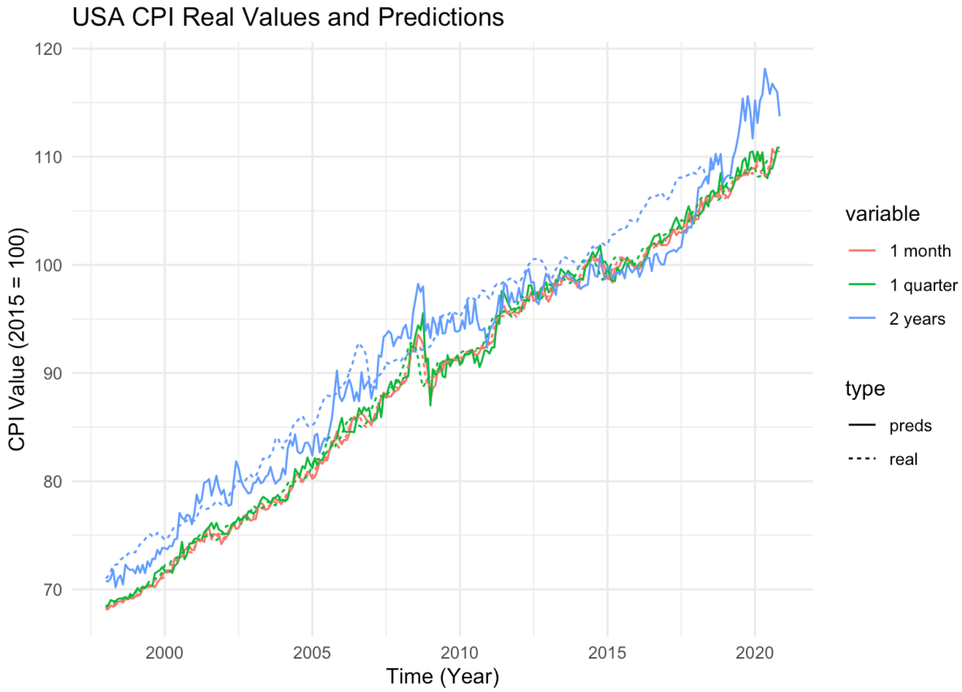
5.5 Updated Rolling Predictions

As for the predictions on the newly filled dataset, we notice that with XGBoost, the predictions are a little more volatile than before. Further, for Canada specifically, they still struggle from 2010 onwards.

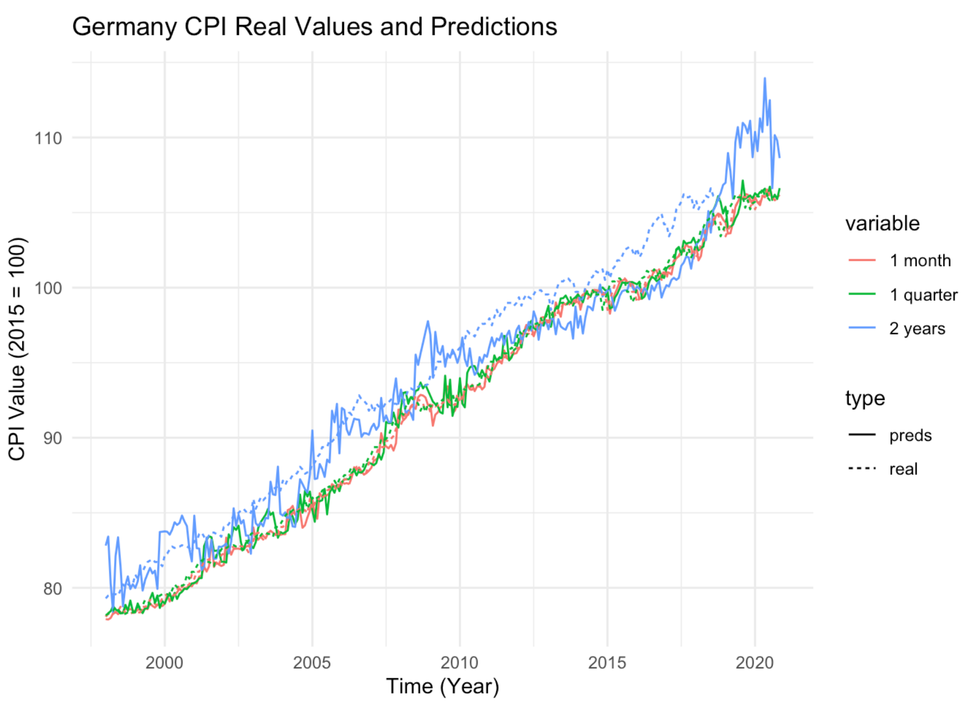
```
ggplot(predsFilled %>% filter(Location == "can", !is.na(value))) + geom_line(aes(x=1980+Time/12, y=value, linetyp
e = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value (2015 = 100)", title = "Canada CPI Real Values
and Predictions") + scale_color_discrete(labels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



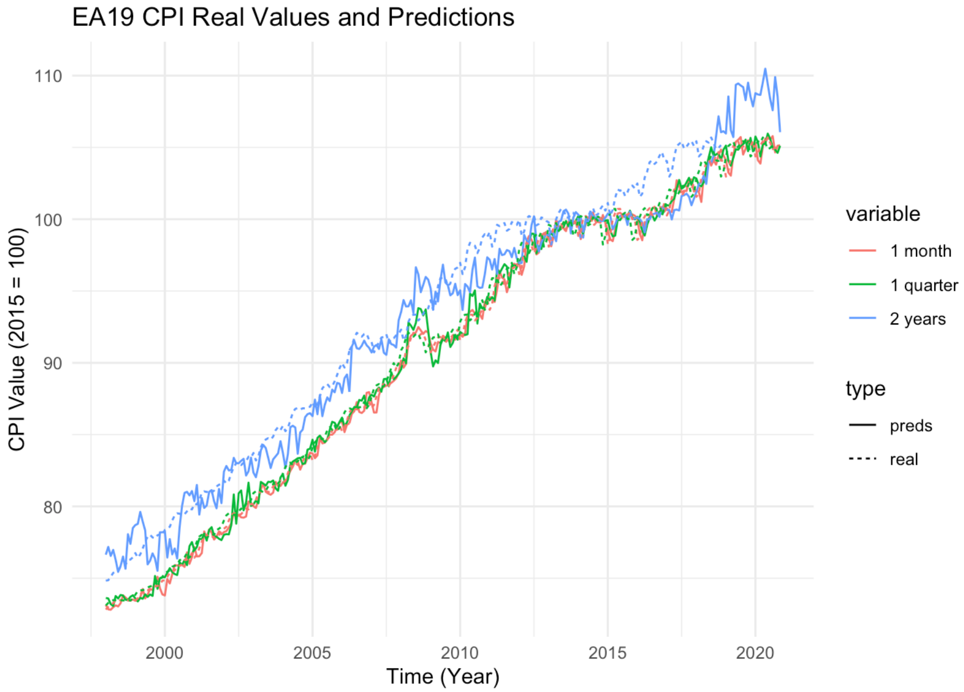
```
ggplot(predsFilled %>% filter(Location == "usa", !is.na(value))) + geom_line(aes(x=1980+Time/12, y=value, linetyp
e = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value (2015 = 100)", title = "USA CPI Real Values and
Predictions") + scale_color_discrete(labels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



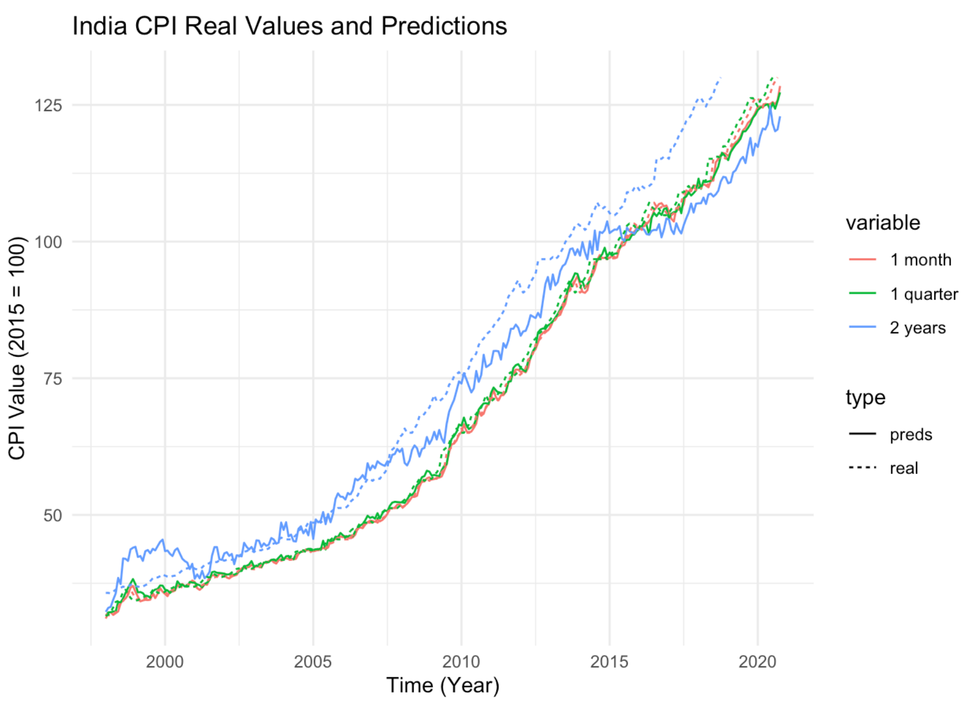

```
ggplot(predsFilled %>% filter(Location == "deu", !is.na(value))) + geom_line(aes(x=1980+Time/12, y=value, linetyp
e = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value (2015 = 100)", title = "Germany CPI Real Values
and Predictions") + scale_color_discrete(labels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



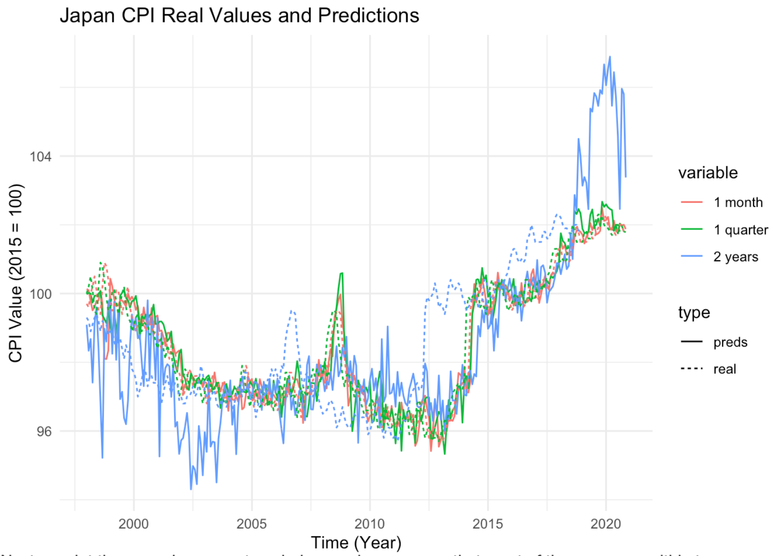
```
ggplot(predsFilled %>% filter(Location == "ea19", !is.na(value))) + geom_line(aes(x=1980+Time/12, y=value, linety
pe = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value (2015 = 100)", title = "EA19 CPI Real Values a
nd Predictions") + scale_color_discrete(labels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```




```
ggplot(predsFilled %>% filter(Location == "ind", !is.na(value))) + geom_line(aes(x=1980+Time/12, y=value, linetyp
e = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value (2015 = 100)", title = "India CPI Real Values a
nd Predictions") + scale_color_discrete(labels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```

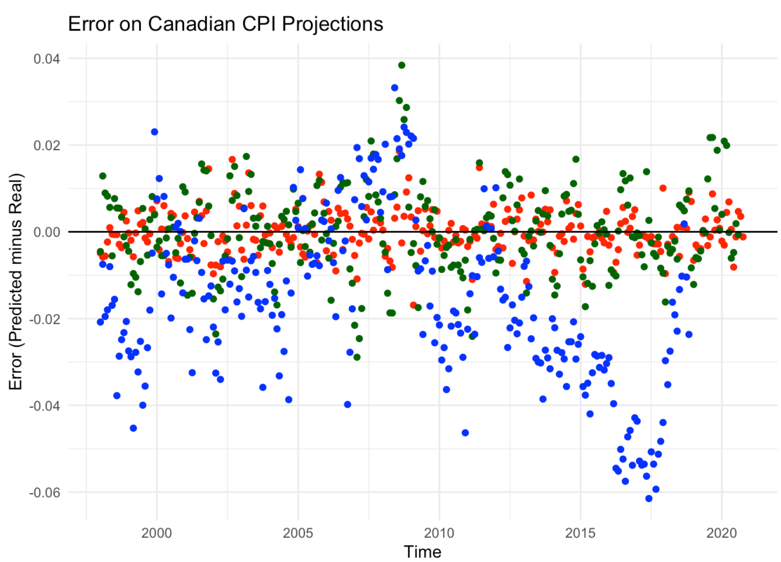


```
ggplot(predsFilled %>% filter(Location == "jpn", !is.na(value))) + geom_line(aes(x=1980+Time/12, y=value, linetyp
e = type, col = variable)) + labs(x="Time (Year)", y = "CPI Value (2015 = 100)", title = "Japan CPI Real Values a
nd Predictions") + scale_color_discrete(labels = c("1 month", "1 quarter", "2 years")) + theme_minimal()
```



Next we plot the errors by percentage below, and we can see that most of the errors are within two percent, except for predictions 2 years out, which generally underpredict after 2010.

```
predsFilledW %>%
  filter(Location == "can") %>%
  ggplot +
  geom_point(aes(x=1980+Time/12, y=(m1_preds-m1_real)/m1_real), col = "red") +
  geom_point(aes(x=1980+Time/12, y=(q1_preds-q1_real)/q1_real), col = "darkgreen") +
  geom_point(aes(x=1980+Time/12, y=(q8_preds-q8_real)/q8_real), col = "blue") +
  labs(x = "Time", y = "Error (Predicted minus Real)", title = "Error on Canadian CPI Projections") +
  geom_hline(aes(yintercept=0)) +
  theme_minimal()
```



Finally, we give our one-quarter-ahead root-mean-squared-errors (in percent). These appear to be similar to the previous results.

```
100*(chgF %>% summarise(m1mse = sqrt(mean(m1c^2, na.rm=TRUE)), q1mse = sqrt(mean(q1c^2, na.rm=TRUE)), q8mse = sqr
t(mean(q8c^2, na.rm=TRUE))))
```

```
##        m1mse     q1mse    q8mse
## 1 0.5032563 0.9921569 2.504202
```

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pages 265–283, 2016.

[2] I. W. Bache, A. S. Jore, J. Mitchell, and S. P. Vahey. Combining VAR and DSGE forecast densities. *Journal of Economic Dynamics and Control*, 35(10):1659–1670, 2011.

[3] Bank of Canada. Understanding the consumer price index.

[4] Bank of Canada. Understanding inflation.

[5] M. Bolhuis and B. Rayner. The more the merrier? a machine learning algorithm for optimal pooling of panel data. *A Machine Learning Algorithm for Optimal Pooling of Panel Data (February 2020). IMF Working Paper*, (20/44), 2020.

[6] L. Breiman et al. Statistical modeling: the two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3):199–231, 2001.

[7] N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge University Press, 2006.

[8] J. Champagne, G.-P. Bellisle, and R. Sekkel. Evaluating the bank of canada staff economic projections using a new database of real-time data and forecasts. Technical report, Bank of Canada Staff Working Paper, 2018.

[9] S. Chen and R. Ranciere. Financial information and macroeconomic forecasts. *International Journal of Forecasting*, 35(3):1160–1174, 2019.

[10] R. Chetty, J. N. Friedman, N. Hendren, M. Stepner, et al. How did covid-19 and stabilization policies affect spending and employment? a new real-time economic tracker based on private sector data. Technical report, National Bureau of Economic Research, 2020.

[11] K. P. Christoffel, G. Coenen, and A. Warne. Forecasting with DSGE models. 2010.

[12] B. De Paoli, A. M. Sbordone, and A. Tambalotti. A Bird's Eye View of the FRBNY DSGE Model. *Liberty Street Economics*, Sep 2014.

[13] M. Del Negro and F. Schorfheide. DSGE model-based forecasting. In *Handbook of economic forecasting*, volume 2, pages 57–140. Elsevier, 2013.

[14] X. Deng and Y. Du. The computation of approximate competitive equilibrium is PPAD-hard. *Information Processing Letters*, 108(6):369–373, 2008.

[15] J. Dorich, M. K. Johnston, R. R. Mendes, S. Murchison, and Y. Zhang. ToTEM II: An updated version of the Bank of Canada's quarterly projection model. Technical report, Bank of Canada, 2013.

[16] W. C. Dudley. Important Choices for the Federal Reserve in the Years Ahead.

[17] K. Engemann. The Fed's Inflation Target: Why 2 Percent?

[18] P. Fenton, S. Murchison, et al. ToTEM: The Bank of Canada's New Projection and Policy-Analysis Model. *Bank of Canada Review*, 2006(Autumn):5–18, 2006.

[19] C. Fulton. Estimating a Real Business Cycle DSGE Model by Maximum Likelihood in Python. 2015.

[20] H. Gerard and K. Nimark. Combining multivariate density forecasts using predictive criteria. 2008.

[21] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT press Cambridge, 2016.

[22] J. T. Harvey. What Actually Causes Inflation.

[23] E. P. Herbst and F. Schorfheide. *Bayesian estimation of DSGE models*. Princeton University Press, 2015.

[24] S. Kelton. *The Deficit Myth: Modern Monetary Theory and How to Build a Better Economy*. Hachette UK, 2020.

[25] Keras. Regularizers.

[26] N. Kocherlakota. Modern Macroeconomic Models as Tools for Economic Policy. *Federal Reserve Bank of Minneapolis*, May 2010.

[27] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30:6402–6413, 2017.

[28] A. Macfie. Uncertainty due to computational approximation in Bayesian inference, 2020.

[29] A. Manstead. The psychology of social class: How socioeconomic status impacts thought, feelings, and behaviour. *The British journal of social psychology*, page 267–291, 2018.

[30] M. C. Medeiros, G. F. Vasconcelos, A. Veiga, and E. Zilberman. Forecasting inflation in a data-rich environment: the benefits of machine learning methods. *Journal of Business & Economic Statistics*, pages 1–22, 2019.

[31] M. H. Pesaran, T. Schuermann, and L. V. Smith. Forecasting economic and financial variables with global VARs. *International journal of forecasting*, 25(4):642–675, 2009.

[32] S. S. Poloz. Models and the Art and Science of Making Monetary Policy, 2017.

[33] R. Sutton and G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[34] S. Verstyuk. Modeling multivariate time series in economics: From auto-regressions to recurrent neural networks. *Available at SSRN 3589337*, 2020.

[35] M. H. Wolters. Evaluating point and density forecasts of DSGE models. *Journal of Applied Econometrics*, 30(1):74–96, 2015.