SPF Code

J. Wren*

January 17, 2024

1 Introduction

The purpose of the SPF code is to restructure the raw data such that computations (e.g., interpersonal disagreement) and regressions can be carried out.

2 Data

2.1 Data Files and Variables

There are two data files – the SPF file and the realized outcomes of six variables of interest in the survey:

1. Annual real GDP growth (PRGDP)
2. Annual GDP price inflation (PRPGDP)
3. Annual-average unemployment rate (PRUNEMP)
4. Annual core CPI inflation (PRCCPI)
5. Annual core PCE inflation (PRCPCE)
6. The probability of a decline in the level of real GDP from one quarter to the next (RECESS)[1]

While I distinguish each variable, no such distinction is made regarding the computations and the majority of regressions previously mentioned.

*Faculty Supervisor: M. Backus

[1]However, the RECESS variable is not included in the analysis.

2.2 SPF Data

Forecasters estimate the probability that the annual-over-annual percent change in each variable (e.g., GDP) falls into various predefined ranges or bins. Unfortunately, for the variables PRGDP and PRPGDP, the ranges and the number of annual forecast horizons change over time. For instance, the number of bins changes from 15, to six, then to 10, and currently to 11. To illustrate the probability bins, the ranges for PRPGDP across two 'time segments' are shown below:

Table 1: Probability Bins for PRPGDP (GDP Prices)

No.	**1992:Q1 to 2013:Q4**	**2014:Q1 to Present**
1	8+	4.0 or more
2	7 to 7.9	3.5 to 3.9
3	6 to 6.9	3.0 to 3.4
4	5 to 5.9	2.5 to 2.9
5	4 to 4.9	2.0 to 2.4
6	3 to 3.9	1.5 to 1.9
7	2 to 2.9	1.0 to 1.4
8	1 to 1.9	0.5 to 0.9
9	0 to 0.9	0.0 to 0.4
10	<0	Will decline

3 Code

We will proceed in the order of the code (see *main.R*), discussing the upshot of each of the main functions. Please note that many tasks are carried out by helper functions that we may not explicitly mention by name.

3.1 Bin outcomes

For each variable in the outcomes file, the year and the realization are provided. We map each realized outcome (via *fn_realized_outcomes_to_bins.R*) to a bin number. For example, the realization of PRGDP in 2003 was 2.8, and we map this to bin number five.

3.2 SPF Restructuring and Cleaning

The SPF data file consists of the year and quarter in which the survey was conducted, an ID corresponding to a forecaster, the industry of the forecaster (not used), and the remaining columns are the probability forecasts across the relevant bins (since the number of bins may not be consistent across time, some columns may not be applicable.). We (via *fn_clean.R*) first stack the bins (and the corresponding probability forecasts), so that for each survey-event, the forecaster is repeated according to the number of bins. Thereafter, many manual changes are made that differ across time, the number of bins, and the variable (being forecasted). For instance, consider the third quarter survey of 1981 and hone in on the forecaster with ID 12. Across the 12 bins in that survey, the probability forecasts should sum to one. However, we see a probability of 0.05 assigned to bins two and four, a probability of 0.90 to bin three, with zeros to bins one, five, and six, but the density projections across bins seven through twelve, also sum to one. This is a consequence of what was previously mentioned – the annual forecast horizon may change across time, where in this survey, the forecasters provided density projections for the current year *and* the next.

3.3 Adding Realizations

After the manual adjustments, we then merge in the corresponding realizations (via *fn_add_outcomes.R*) and effectively create a degenerate forecast. For concreteness, consider PRPGDP in 1997 for forecaster 420. Like before, we have projections for two years on a quarterly basis (quarter three in 1996 is missing) across 10 bins. To aid the merge, we make use of a time variable, where we take time to be the survey 'midpoint' (so, Q2 of 1997 corresponds to time 1997.375). Thus, the first density projection is at time 1996.125 and the last is at 1997.875. We then add the realization as a 'forecast' with time equal to 1997.999 (and with the quarter set to five) but with all the probability on the correct bin.

3.4 Bin numbers to Bin Endpoints

Next, we map the bin numbers (not really useful) to bin endpoints. There is one assumption here that is likely worth discussion. We need to assign a value to each bin. The default is to take the midpoint, but for the left and rightmost bins, that isn't viable. The current implementation simply adds one to the left (right) endpoint for the rightmost (leftmost) bin. So for the e.g., 10+ bin, the value is 11 and for the e.g., less than -3, it is -4.

3.5 Computations

Next, (in *fn_ind_computations.R*)[2] we compute various measures such as the expected value, variance, and disagreement. Firstly, we compute an expected value for each forecaster for a

[2]The 'ind' corresponds to 'individual', meaning at the level of each individual forecaster.

single time/event, where we simply calculate a weighted average across the bins, weighted by the probability forecasts. The variance computation is similar, but we also have to add the within-bin variance, where we assume a uniform distribution within each bin.[3] Similar to the bin values of the 'bounding' bins previously noted, we set the size of the bounding bins to one.

I also implemented a 'smooth' version, where I smooth out the forecaster's predictions (i.e., less discrete). For simplicity, I will describe the steps through an example. In Q4 of 1968 (the first survey), forecaster one put probability of 0.10, 0.30, and 0.60 on the bins [3, 4), [2, 3), and [1, 2) respectively, and zero on the other 12 bins. I then replicate each row q[4] times, where importantly, q is increasing in the density projection. In this example, the rows associated with the probabilities, 0.10, 0.30, and 0.60, were replicated 50, 150, and 300 times. Next, for each bin, we linearly interpolate the bin values. Subsequently, we fit a density over the finely partitioned bins. Lastly, we (numerically) integrate over the density to compute the expected value (EV) and variance.

The expected value then allows us to compute the squared error for each forecaster at a point in time for a given event, where the error is simply the difference between the EV and the realization. We also compute the disagreement (in terms of EV) between forecasters, which is simply the variance. Similarly, we calculate the disagreement excluding the i^{th} forecaster. Lastly, turning to the variance, we compute the deviation between the squared error and the predicted variance.

3.5.1 Aggregating

Next, (via *fn_agg_computations.R*) we aggregate the data and again calculate various measures (previously discussed). The aggregation is performed at the event-time-bin level, where we average over individuals, creating an 'average' forecaster.

3.6 Analysis and Figures

Lastly (via *fn_analysis.R*), we run the appropriate regressions and generate the calibration plots, both 'smoothed' and 'binned' at the individual and average forecaster levels. We will limit our discussion to what exactly is meant by 'smoothed', but first we need to set some context. We will first address what is meant by calibration. The idea is that we

[3] The way this was previously computed appears to be incorrect.

[4] Formally, q is defined as $500p \cdot (\text{binh - binl})$, where p is the probability forecast of a particular bin and binh and binl are the associated bin endpoints. For numerical stability, I bound q below at 10.

have predicted probabilities (in most cases, predicted by a model) and true or empirical probabilities. Calibration refers to better aligning these distributions. A calibration curve is a diagnostic tool, where the true or observed relative frequency goes on the y-axis and the predicted probability frequency on the x-axis. The traditional calibration plot is a binned scatter plot. We (via *fn_calibration.R*) first cut or bin the probabilities (in our case, we have nine bins). Then we create indicator variables matching each probability forecast to the appropriate bin. E.g., in 2006, Q3, forecaster 506, put 0.90 probability of PRGDP landing in [3,4) – which happens to align with the ninth bin, where we place a one. The forecaster put the remaining probability, 0.10, in the bin [2,3), which was the correct bin (the realization was 2.8). Next, we create a binary variable, that is equal to one if the realization lands in that particular bin. Thereafter, we regress the binary 'true bin' indicator on the nine dummy variables, clustering by event and forecaster (we only cluster by event when aggregating across forecasters). The purpose of the regression is to get error bars for the calibration scatter plot, where the error bars bound the binned average probability forecast. Finally, we return to smoothing. The idea is to 'smooth out' the true bin indicator. The approach taken is quite simple: a curve is fit through the data points (probability forecast, true bin indicator) via a spline. The spline passes through or near these points in a way that minimizes the overall curvature of the line, resulting in a smooth calibration curve.

References

[1] Federal Reserve Bank of Philadelphia. *Survey of Professional Forecasters Documentation*. Available at: https://www.philadelphiafed.org/-/media/frbp/assets/surveys-and-data/survey-of-professional-forecasters/spf-documentation.pdf?la=en&hash=F2D73A2CE0C3EA90E71A363719588D25.