Spurious Sentience

On the Unsurprising Finding of Patterns in Latent Spaces

Patrick Altmeyer

2024-01-10

For the slides see here.

We humans are prone to seek patterns everywhere. Meaningful patterns have often proven to help us make sense of our past, navigate our presence and predict the future. Our society is so invested in finding patterns that today it seems we are more willing than ever to outsource this task to an Artificial Intelligence (AI): an omniscient oracle that leads us down the right path. Unfortunately, history has shown time and again that patterns are double-edged swords: if we attribute the wrong meaning to them, they may lead us nowhere at all, or worse, they may lead us down the dark roads.

In statistics, misleading patterns are referred to as **spurious relationships**: purely associational relationships between two or more variables that are not causally related to each other at all. The world is full of these and as good as we as species may be at recognizing patterns, we typically have a much harder time discerning spurious relationships from causal ones. Despite new and increased momentum in scientific fields concerned with causal inference and discovery, I am also willing to go out on a limb and claim that we are not about to finally reach the top of Judea Pearl's Causal Ladder through the means of Causal AI.

I agree with the premise that in a world full of spurious relationships, causal reasoning is our only remedy. But I am very skeptical of claims that AI will magically provide that remedy. This leads me to the title and topic of this post: **spurious sentience**—patterns exhibited by artificial intelligence that may hint at sentience but are just reflections of the data used to train them. The article is written in response to a recent paper that extracts a 'world model' from Llama 2—a popular open-source large language model (LLM)—using mechanistic interpretability (Gurnee and Tegmark 2023). In light of these findings, one of the authors, Max Tegmark, was quick to claim on social media that "No, LLM's aren't mere stochastic parrots [...]".

Since this is an opinionated post, I feel that I should start with a few disclaimers:

1. I take no issue with the methodological ideas that form the foundation of the article in question: on the contrary, I think that mechanistic interpretability is an interesting and important toolkit that can help us better understand the intrinsics and behavior of opaque artificial intelligence.
2. The visualizations are intriguing, the code is open-sourced and the findings are interesting.
3. I am surprised that people are surprised by the findings: if we agree that LLMs exhibit strong capabilities that can only be connected to the patterns observed in the data they were trained with, then where exactly should we expect this information to be stored if not in the parameters of the model?[1]
4. I therefore do take issue with the way that these findings are being overblown by people with clout. Perhaps the parrot metaphor should not be taken too literally either, but if anything the paper's findings seem to support the notion that LLMs are remarkably capable of memorizing and regurgitating explicit and implicit knowledge contained in text.

Patterns in Latent Spaces and How to Find Them

To support my claim that observing patterns in latent spaces should not generally surprise us, we will now go through a couple of simple examples. To illustrate further that this phenomenon is neither surprising nor unique to the field of computer science, I will draw on my background in economics and finance in this section. We will start with very simple examples to demonstrate that even small and simple models can learn meaningful representations of the data. The final example in **?@sec-ex-llm** is a bit more involved and closer in spirit to the experiments conducted by Gurnee and Tegmark (2023). As we go along, we will try to discuss both the benefits and potential pitfalls of finding patterns in latent spaces.

Example: Principal Component Analysis

In response to the claims made by Tegmark, numerous commentators on social media have pointed out that even the simplest of models can exhibit structure in their latent spaces. One of the most popular and illustrative examples I remember from my time at the Bank of England is yield curve decomposition through PCA. The yield curve is a popular tool for investors and economists to gauge the health of the economy. It plots the yields of bonds against their maturities. The slope of the yield curve is often used as a predictor of future economic activity: a steep yield curve is associated with a growing economy, while a flat or inverted yield curve is associated with a contracting economy.

To understand this better, let us go on a quick detour into economics and look at actual yield curves observed in the US during the Global Financial Crisis (GFC). Figure 1a shows the yield

[1] I would be very surprised—concerned even—if our search for patterns in latent spaces of capable LLMs revealed nothing at all.

curve of US Treasury bonds on 27 February 2007, which according to CNN was a "brutal day on Wall Street".[2] This followed reports on the previous day of former Federal Reserve Chairman Alan Greenspan's warning that the US economy was at risk of a recession. The yield curve was inverted with a sharp negative spread between the 10-year and 3-month yields, indicative of the market's expectation of a recession.

Figure 1b shows the corresponding yield curve during the aftermath of the GFC on 20 April 2009. On that day the influential Time Magazine reported that the "Banking Crisis is Over". The yield curve was steeply sloped with a positive spread between the 10-year and 3-month yields, indicative of the market's expectation of a recovery. The overall level of the yield curve was still very low though, indicative of the fact that US economy had not fully recovered at that point.

Of course, US Treasuries are not the only bonds that are traded in the market. To get a more complete picture of the economy, analysts might therefore be interested in looking at the yield curves of other bonds as well. In particular, we might be interested in predicting economic growth based on the yield curves of many different bonds. The problem with that idea is that it is cursed by high dimensionality: we would end up modelling a single variable of interest (economic growth) with a large number of predictors (the yields of many different bonds). To deal with the curse of high dimensionality it can be useful to decompose the yield curves into sets of principal components.

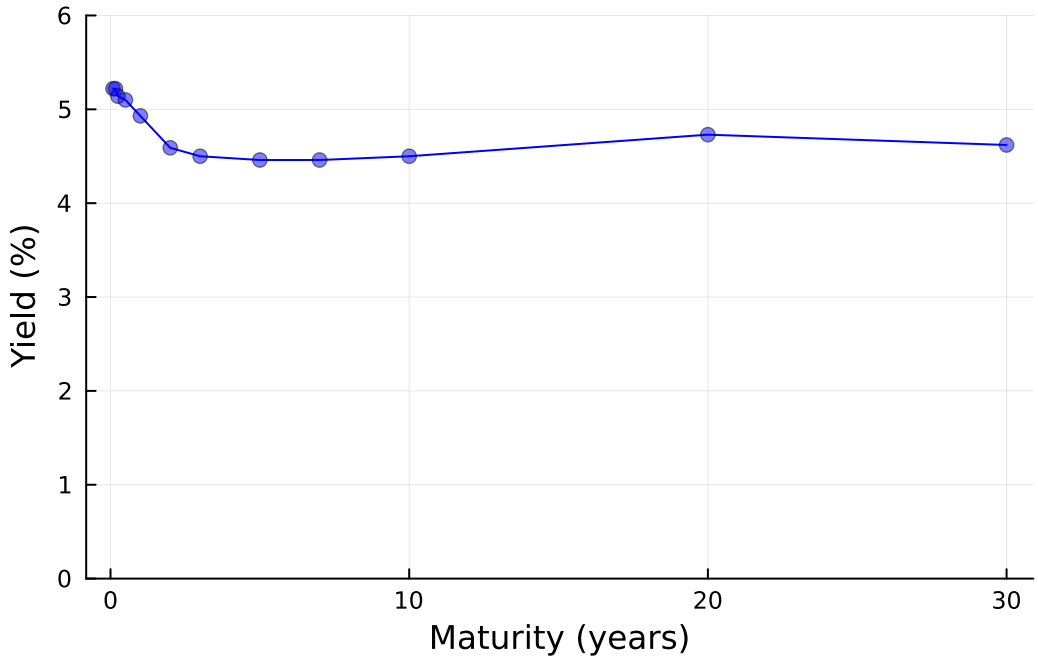
To compute the principal components we can decompose the matrix of yields $\mathbf{Y}$ into a product of its singular vectors and values: $\mathbf{Y} = \mathbf{U}\Sigma\mathbf{V}'$. I will not go into the details here, because Professor Gilbert Strang has already done a much better job than I ever could in his Linear Algebra lectures. To put this into the broader context of the article, however, let us simply refer to $\mathbf{U}$, Σ and $\mathbf{V}'$ as latent embeddings of the yield curve (they are latent because they are not directly observable).

The top panel in Figure 2 shows the first two principal components of the yield curves of US Treasury bonds over time. Vertical stalks indicate the key dates during the onset and aftermath of the crisis, which we discussed above. For both components, we can observe some marked shifts between the two dates - but can we attribute any meaning to these shifts? It turns out we can: for comparison, the bottom panel in Figure 2 shows the average level and spread of the yield curves over time. The first principal component is strongly correlated with the level of the yield curve, while the second principal component is strongly correlated with the spread of the yield curve. To put it in AI-lingo:

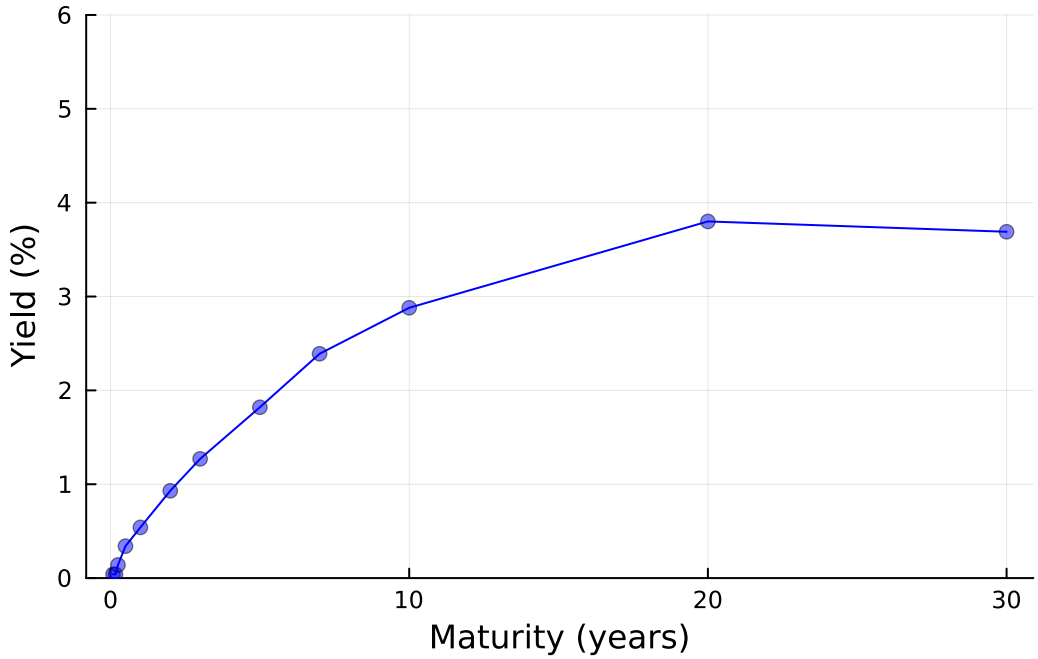
> The estimated latent embeddings of the yield curve are characterized by patterns observed in the data.

Not convinced? Let us use $\mathbf{Y} = \mathbf{U}\Sigma\mathbf{V}'$ in true autoencoder fashion to reconstruct yield curves from principal components. Let z_1 denote the first principal component and consider the following: we keep all other $M - 1$ principal components fixed at zero where M denotes the

[2]The data is taken from the US Department of the Treasury.



(a) Onset of GFC: 27 February 2007.



(b) Aftermath of GFC: 20 April 2009.

Figure 1: Yield curve of US Treasury bonds.

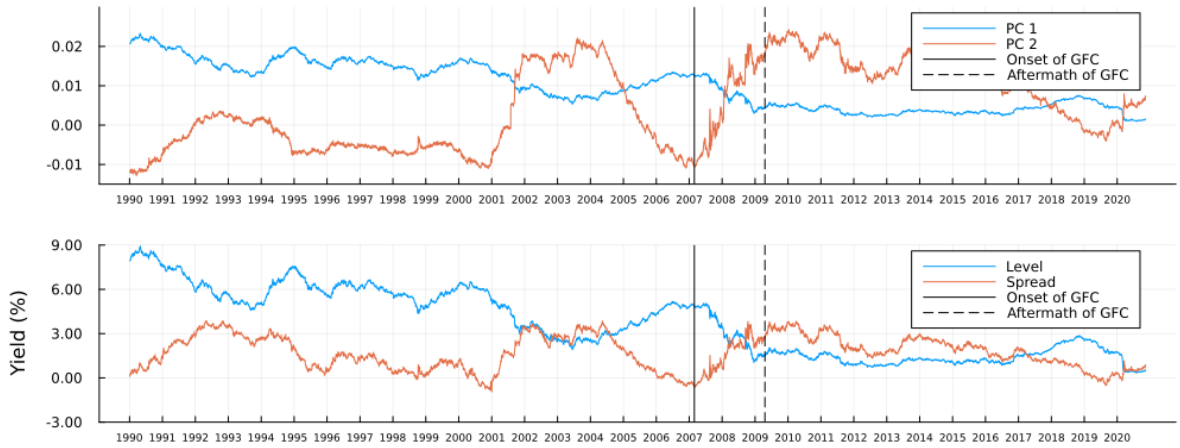


Figure 2: Comparison of latent embeddings and observed data of the US Treasury yield curve.

total number of maturities; next we traverse the latent space by varying the value of z_1 over a fixed grid of length K each time storing the full vector $\mathbf{z}$; finally, we vertically concatenate the vectors and end up with a matrix $\mathbf{Z}$ of dimension $(K \times M)$. To reconstruct yields, we simply multiply Z by the singular values and right singular vectors: $\mathbf{Y} = \mathbf{Z}\Sigma\mathbf{V}'$.

?@fig-pca-anim shows the result of this exercise in the left panel. As we can see, our generated yield curves shift vertically as we traverse the latent space. The right panel of **?@fig-pca-anim** shows the result of a similar exercise, but this time we keep the first principal component fixed at zero and vary the second principal component. This time the slope of our generated yield curves shifts as we traverse the latent space.

Example: Deep Learning

So far we have considered simple matrix decomposition. You might argue that principal components are not really latent embeddings in the traditional sense of deep learning. To address this, let us now consider a simple deep-learning example. Our goal will be to not only predict economic growth from the yield curve but also extract meaningful features at the same time. In particular, we will use a neural network architecture that allows us to recover a compressed latent representation of the yield curve.

Data

To estimate economic growth we will rely on a quarterly series of the real gross domestic product (GDP) provided by the Federal Reserve Bank of St. Louis. The data arrives in terms of levels of real GDP. In order to estimate growth, we will transform the data into log differences. Since our yield curve data is daily, we will need to aggregate it to the quarterly frequency. To

do this, we will simply take the average of the daily yields for each maturity. We will also standardize yields since deep learning models tend to perform better with standardized data. Since COVID-19 was a huge structural break, we will also filter out all observations after 2018. Figure 3 shows the pre-processed data.

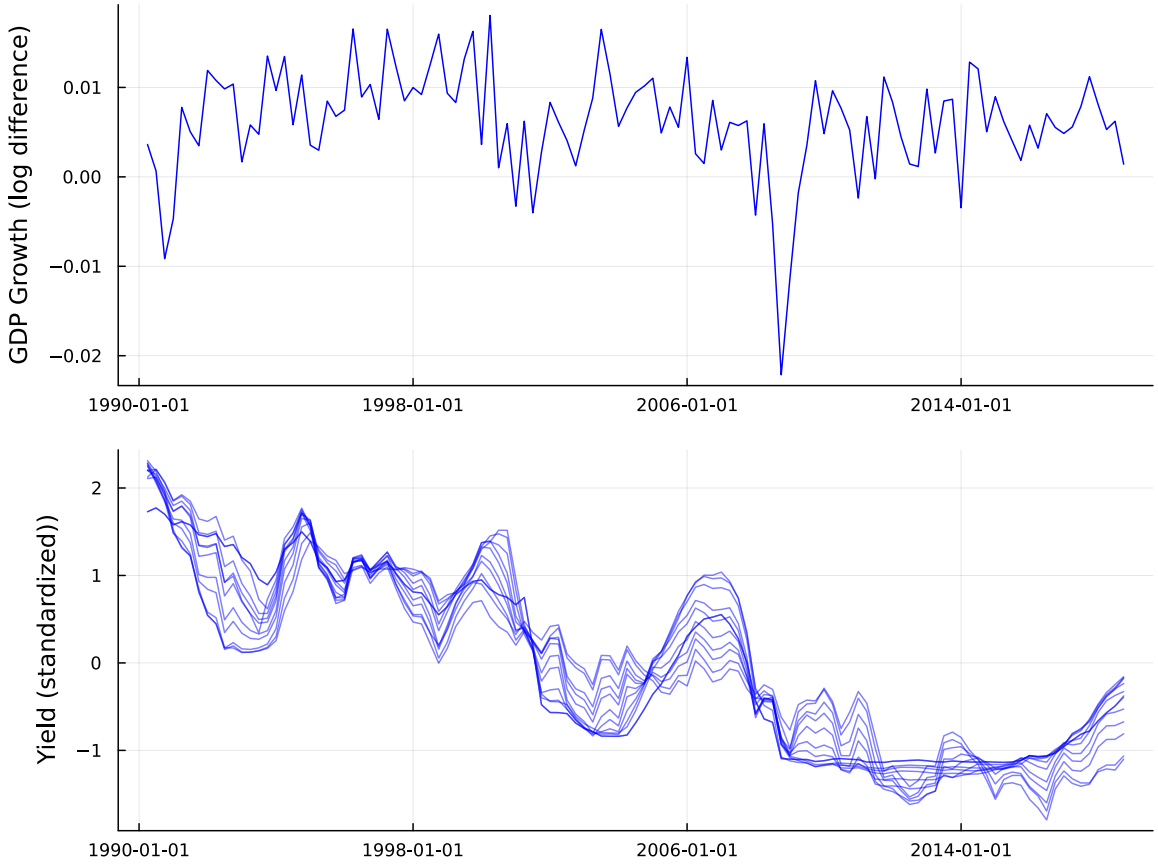


Figure 3: GDP growth and yield curve data.

Model

Let G_t denote growth and $\mathbf{Y}_t$ denote the yield curve at time t. Then we are interested in a model for G_t conditional on $\mathbf{Y}_t$. Let θ denote our model parameters then formally we are interested in maximizing the likelihood $p_\theta(G_t|\mathbf{Y}_t)$. To do this, we will use a simple autoencoder architecture that is illustrated in Figure 4. The encoder will consist of a single fully connected hidden layer with 32 neurons and a hyperbolic tangent activation function. The bottleneck layer connecting the encoder to the decoder is a fully connected layer with 6 neurons. This is the compressed latent representation of the yield curve mentioned above. The decoder will consist of two fully connected layers each with a hyperbolic tangent activation function: the first layer

will consist of 32 neurons and the second layer will have the same dimension as the input data. The output layer will consist of a single neuron with a linear activation function. The model will be trained using the mean squared error loss function and the Adam optimizer.

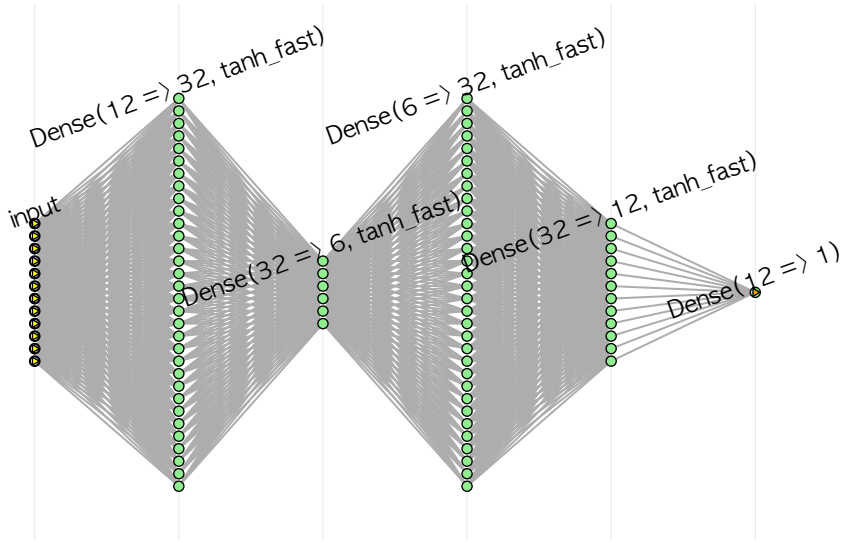


Figure 4: Model architecture.

Linear Probe

The results are shown in Figure 5. The top panel shows the actual GDP growth and fitted values from the autoencoder model. We observe that the model captures the relationship between economic growth and the yield curve reasonably well. As discussed above, we also know that the relationship between economic growth and the yield curve is characterized by two main factors: the level and the spread. Since the model itself is fully characterized by its parameters, we would expect that these two important factors are reflected somewhere in the latent parameter space.

The bottleneck layer seems like a good place to start looking. To get the latent embeddings $\mathbf{A}_t$ at time t, we simply pass the yield curve data $\mathbf{Y}_t$ through the encoder. Next, we follow Gurnee and Tegmark (2023) and use a linear probe to regress the observed yield curve factors on the latent embeddings. Let F_t denote the vector containing the two factors of interest in time t: f_t^{spread} and f_t^{level}. Formally, we are interested in the following regression model: $p_w(F_t|\mathbf{A}_t)$ where w denotes the regression parameters. Following Gurnee and Tegmark (2023), we use Ridge regression with λ set to 0.1. Using the estimated regression parameters $\hat{w}$ we can then predict the yield curve factors from the latent embeddings: $\hat{F}_t = \hat{w}'\mathbf{A}_t$.

The results of this experiment are shown in the bottom panel of Figure 5. Solid lines show the observed yield curve factors over time, while dashed lines show predicted values. We find that the latent embeddings predict the two yield curve factors reasonably well, in particular the spread. To put this in true hype-lingo:

> Not just a parrot! Our autoencoder neural network has an implicit understanding of the yield curve factors and their relationship with economic growth. It's sentient indeed!

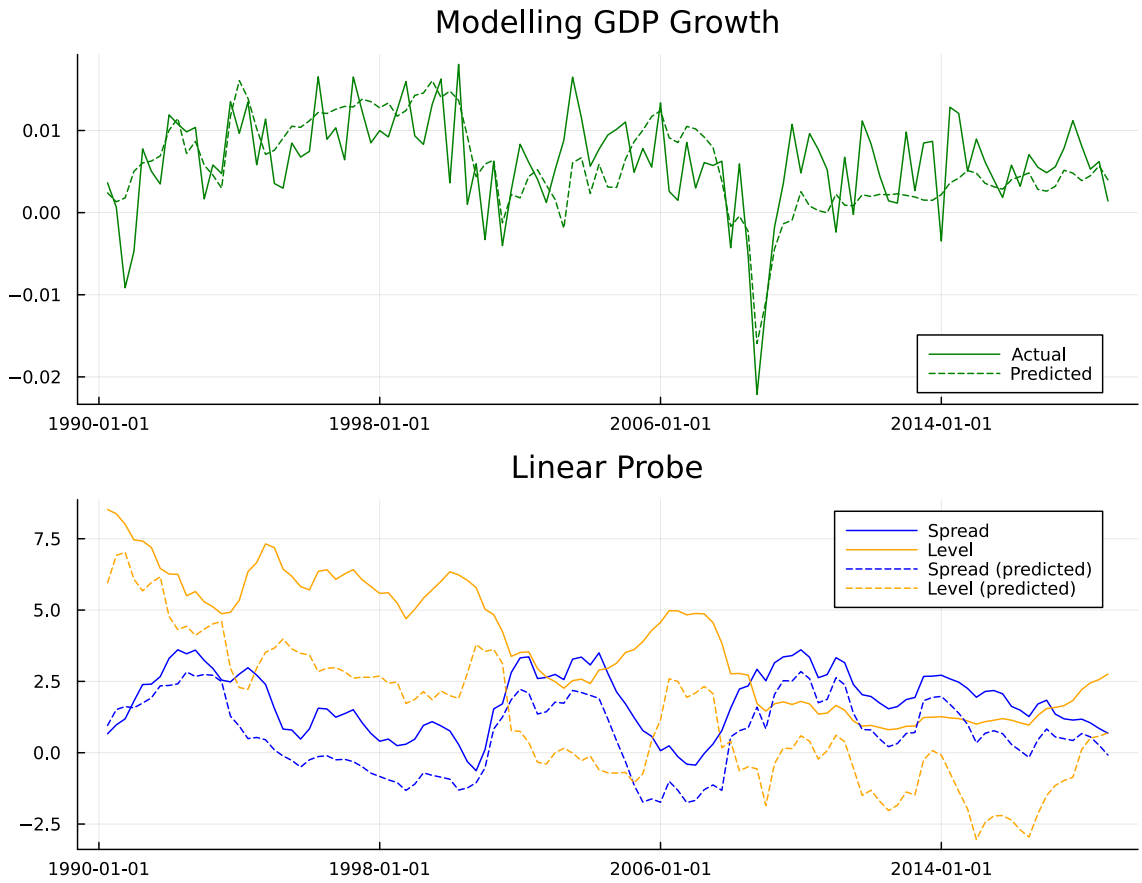


Figure 5: The top panel shows the actual GDP growth and fitted values from the autoencoder model. The bottom panel shows the observed average level and spread of the yield curve (solid) along with the predicted values from the linear probe based on the latent embedding (dashed).

Ok, but truly what's the point?

The finding is not surprising but it is still interesting. In the context of mechanistic interpretabil-

ity, it demonstrates that the black-box model has evidently learned plausible explanations for the data. Beyond that, in this particular example, the patterns in the latent space that we have just uncovered might actually be useful for downstream tasks. An interesting idea could be to use the latent embeddings as features in a more traditional and interpretable econometric model. To demonstrate this, let us consider a simple linear regression model for GDP growth. We will compare the performance of the following models: (1) regressing growth G_t on the latent embedding A_t, (2) regression growth on the best subset of latent embeddings A_t, (3) regressing growth on lagged growth, (4) regressing growth on lagged growth and the observed yield curve factors F_t, (5) regressing growth on the best subset of yield curve factors F_t, and, finally, (5) regressing growth on lagged growth and the best subset of latent embeddings A_t.

The results are shown in the table below. The key finding of interest is that the coefficients on the latent embeddings in model (5) are statistically significant. This suggests that the latent embeddings contain information that is useful for predicting GDP growth.

	y				
	(1)	(2)	(3)	(4)	(5)
(Intercept)	0.009***	0.006***	0.004***	0.002	0.004***
	(0.002)	(0.001)	(0.001)	(0.002)	(0.001)
a1	0.007				
	(0.010)				
a2	0.022**				
	(0.008)				
a3	-0.051*				
	(0.022)				
a4	0.017				
	(0.010)				
a5	0.020*				
	(0.009)				
a6	0.046**	0.011***			0.008*
	(0.015)	(0.003)			(0.003)
yl			0.398***	0.385***	0.344***
			(0.087)	(0.089)	(0.088)
spread				0.000	
				(0.001)	
level				0.000	
				(0.000)	
N	115	115	114	114	114
R^2	0.169	0.093	0.158	0.168	0.203

Example: Large Language Model

To round up this section, we will jump back on the hype train and consider an example involving an LLM. In particular, we will closely follow the approach in Gurnee and Tegmark (2023) and apply it to a novel financial dataset: the *Trillion Dollar Words* dataset introduced by Shah, Paturi, and Chava (2023). The dataset contains a curated selection of sentences formulated by central bankers of the US Federal Reserve and communicated to the public in speeches, meeting minutes and press conferences. The authors of the paper use this dataset to train LLMs to classify sentences as either 'dovish', 'hawkish' or 'neutral'. To this end, they first manually annotate a subsample of the available data and then fine-tune various foundation models. Their model of choice, *FOMC-RoBERTa* (a fine-tuned version of RoBERTa (Liu et al. 2019)), achieves an F_1 score of around > 0.7 for the classification task. To illustrate the potential usefulness of the learned classifier, they use predicted labels for the entire dataset to compute an ad-hoc, count-based measure of 'hawkishness'. They then go on to show that this measure correlates with key economic indicators in the expected direction: when inflationary pressures rise, the measured level of 'hawkishness' increases as central bankers need to raise interest rates to bring inflation back to target.

Linear Probes

Instead of computing a measure based on predicted labels, we can use linear probes to assess if the fine-tuned model has learned associative patterns between central bank communications and key economic indicators. To this end, I have further pre-processed the data provided by Shah, Paturi, and Chava (2023) and used their proposed model to compute layer-wise embeddings for all available sentences. I have made these available and easily accessible through a small Julia package: TrillionDollarWords.jl. For each layer, I have then computed linear probes on two inflation indicators—the Consumer Price Index (CPI) and the Producer Price Index (PPI)—as well as US Treasury yields at different levels of maturity. To mitigate issues related to over-parameterization, I follow the recommendation in Alain and Bengio (2018) to first reduce the dimensionality of the embeddings each time. In particular, linear probes are restricted to the first 128 principal components of the embeddings of each layer.

Detailed results of this experiment will be released in an upcoming paper. Figure 6 highlights the result for the linear probe on the CPI. The chart shows various performance measures plotted against *FOMC-RoBERTa*'s n-th layer. Shaded areas show the variation across cross-validation folds, where we have used an expanding window approach to split the time series. To avoid look-ahead bias, we run PCA separately for each training set. Results for the linear probes are shown along with results for autoregressive models (AR) used as our baseline.

The first column shows the correlations between model predictions and observed inflation. One can observe that this correlation is strictly positive for the baseline models and the linear probes. Consistent with the findings in Gurnee and Tegmark (2023) and Alain and Bengio (2018), we also observe that this correlation tends to be higher for layers near the end of the

transformer model. Those layers produce predictions that are more positively correlated with observed inflation than those produced by the corresponding autoregressive models.

Similarly, we observe that the (root) mean squared error of the linear probe is largely on par with the baseline. The average prediction error is gradually reduced as we move along the horizontal axis, which is again indicative of the notion that layers closer to the end of the neural network have distilled more useful representations.

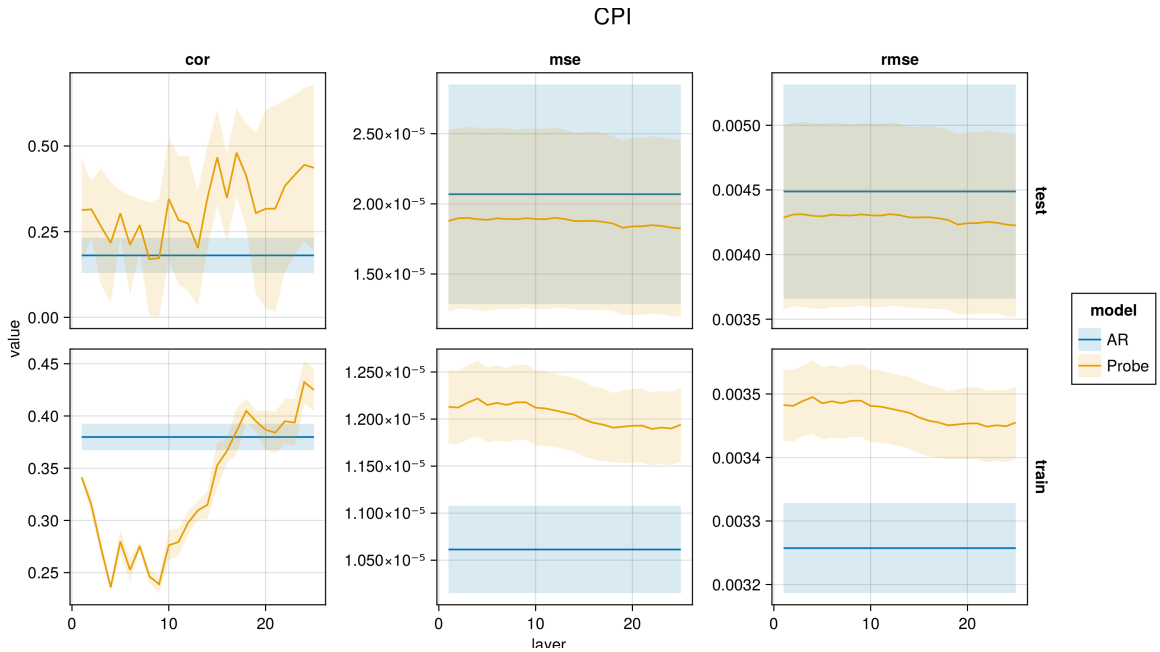


Figure 6: Various performance measures (correlation and (root) mean squared error) plotted against *FOMC-RoBERTa*'s n-th layer. Shaded areas indicate variation across cross-validation folds, where we have used an expanding window approach to split the time series. Results for the linear probes are shown along with results for autoregressive models (AR). The optimal lag length was determined using the Bayes Information Criterium.

Stochastic Parrots After All?

These results from the linear probe shown in Figure 6 are certainly not unimpressive: even though *FOMC-RoBERTa* was not explicitly trained to uncover associations between central bank communications and the level of consumer prices, it appears that the model has distilled representations that can be used to predict inflation. It is worth pointing out here that this model is substantially smaller than the models tested in Gurnee and Tegmark (2023). This begs the following question:

> Have we uncovered further evidence that LLMs "aren't mere stochastic parrots"? Has *FOMC-RoBERTa* developed an intrinsic understanding of the economy just by 'reading' central bank communications?

Personally, I am having a very hard time believing this. To argue my case, I will now produce a counter-example demonstrating that, if anything, these findings are very much in line with the parrot metaphor. The counter-example is based on the following premise: if the results from the linear probe truly were indicative of some intrinsic understanding of the economy, then the probe should not be sensitive to random sentences that are most definitely not related to consumer prices.

To test this, I select the best-performing probe trained on the final-layer activations to predict changes in the CPI. I then make up sentences that fall into one of these four categories: *Inflation/Prices* (IP)—sentences about price inflation, *Deflation/Prices* (DP)—sentences about price deflation, *Inflation/Birds* (IB)—sentences about an inflation in the number of birds and *Deflation/Birds* (DB)—sentences about a deflation in the number of birds. A sensible sentence for category DP, for example, could be: "It is essential to bring inflation back to target to avoid drifting into deflation territory.". Analogically, we could construct the following sentence for the DB category: "It is essential to bring the numbers of doves back to target to avoid drifting into dovelation territory.".

In light of the encouraging results for the probe in Figure 6, we should expect the probe to predict higher levels of inflation for activations for sentences in the IP category than for sentences in the DP category. If this was indicative of true intrinsic understanding, we would not expect to see any significant difference in predicted inflation levels for sentences about birds, independent of whether or not their numbers are increasing. More specifically, we would not expect the probe to predict values for sentences about birds that are substantially different from the values it can be expected to predict when using actual white noise as inputs.

To get to this last point, I also generate many probe predictions for samples of noise. Let $f : \mathcal{A}^k \mapsto \mathcal{Y}$ denote the linear probe that maps from the k-dimensional space spanned by k first principal components of the final-layer activations to the output variable of interest (CPI growth in this case). Then I sample $\varepsilon_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}^{(k\times k)})$ for $i \in [1, 1000]$ and compute the sample average. I repeat this process 10000 times and compute the median-of-means to get an estimate for $\mathbb{E}[f(\varepsilon)] = \mathbb{E}[y|\varepsilon]$, that is the predicted value of the probe conditional on white noise.

Next, I propose the following hypothesis test as a minimum viable testing framework to assess if the probe results (may) provide evidence for an actual understanding of key economic relationships learned purely from text:

Proposition 0.1 (Parrot Test)**.**

- H0 *(Null* WEAK*): The probe never predicts values that are statistically significantly different from* $\mathbb{E}[f(\varepsilon)]$

- H1 *(Stochastic Parrots* RISKIER*): The probe predicts values that are statistically significantly different from* $\mathbb{E}[f(\varepsilon)]$ *for sentences in all categories (IP,DP,IB,DB).*
- H2 *(More than Mere Stochastic Parrots* RISKIEST*): The probe predicts values that are statistically significantly different from* $\mathbb{E}[f(\varepsilon)]$ *for sentences in categories IP and DP, but not for sentences in IB and DB.*

To be clear, if in such a test we did find substantial evidence in favour of rejecting both *HO* and *H1*, this would not automatically imply that *H2* is true. But to even continue investigating if based on having learned meaningful representation the underlying LLM is more than just a parrot, it should be able to pass this simple test.

In this particular case, Figure 7 demonstrates that we find some evidence to reject *H0* but not *H1* for *FOMC-RoBERTa*. The median linear probe predictions for sentences about inflation and deflation are indeed substantially higher and lower, respectively than for random noise. Unfortunately, the same is true for sentences about the inflation and deflation in the number of birds, albeit to a somewhat lower degree.

I should note that the number of sentences in each category is very small here (10), so the results in Figure 7 cannot be used to establish statistical significance. That being said, even a handful of convincing counter-examples should be enough for us to seriously question the claim that results from linear probes provide evidence against the parrot metaphor. In fact, if there's even a single sentence for which any

> Strong alternative hypothesis to address own confirmation bias Articulating a test to better differentiate

References

Alain, Guillaume, and Yoshua Bengio. 2018. "Understanding Intermediate Layers Using Linear Classifier Probes." https://arxiv.org/abs/1610.01644.

Gurnee, Wes, and Max Tegmark. 2023. "Language Models Represent Space and Time." https://arxiv.org/abs/2310.02207.

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. "RoBERTa: A Robustly Optimized BERT Pretraining Approach." https://arxiv.org/abs/1907.11692.

Shah, Agam, Suvan Paturi, and Sudheer Chava. 2023. "Trillion Dollar Words: A New Financial Dataset, Task & Market Analysis." https://arxiv.org/abs/2305.07972.

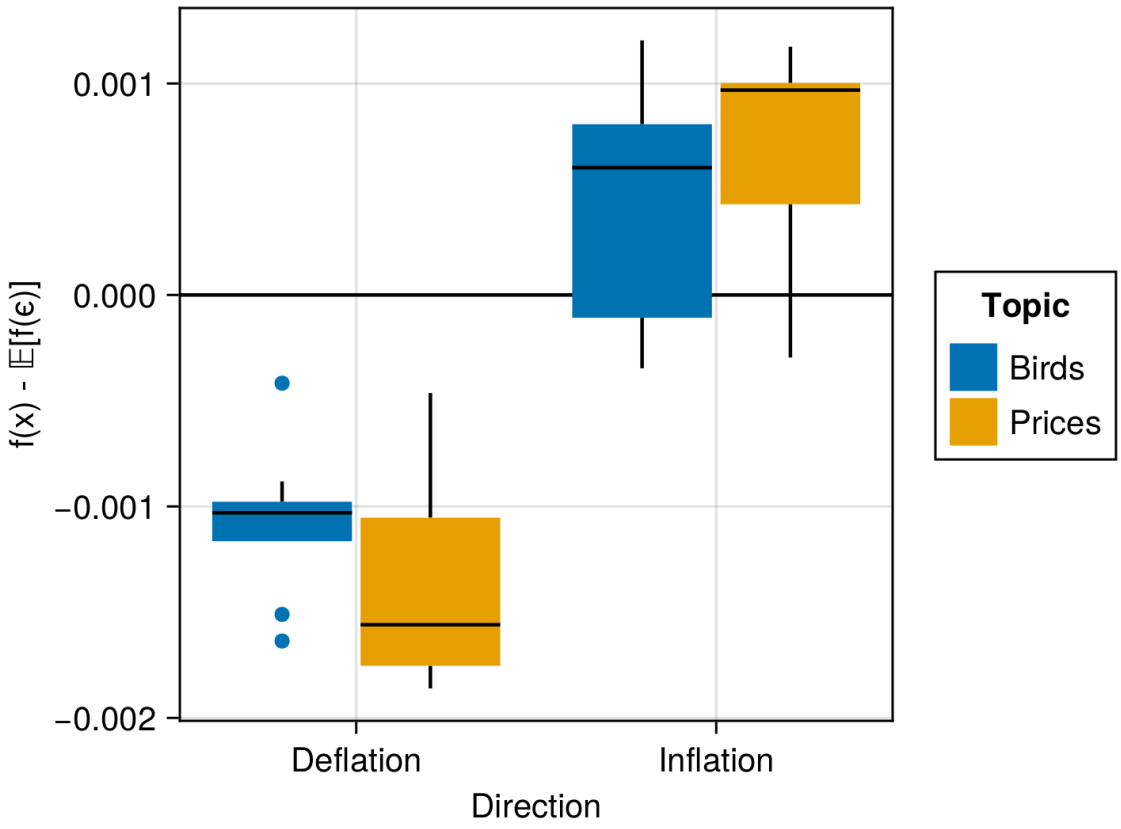


Figure 7: Probe predictions for sentences about inflation of prices (IP), deflation of prices (DP), inflation of birds (IB) and deflation of birds (DB). The vertical axis shows predicted inflation levels subtracted by the average predicted inflation level for the whole sample period (train and test set).