

How to Learn and Teach Economics with Large Language Models, including GPT

Tyler Cowen
Alex Tabarrok

March 17, 2023

Abstract: GPTs, such as ChatGPT and Bing Chat, are capable of answering economics questions, solving specific economic models, creating exams, assisting with research, generating ideas, and enhancing writing, among other tasks. This paper highlights how these innovative tools differ from prior software and necessitate novel methods of interaction. By providing examples, tips, and guidance, we aim to optimize the use of GPTs and LLMs for learning and teaching economics effectively.

I. Introduction

You can learn economics from a textbook, from friends, from YouTube, from private tutors, from running a business, and now from GPTs and the broader category of Large Language Models (LLMs). This essay will guide you in learning and doing economics using these new tools.

If you have played around with any of the latest GPTs, you already understand some of the potential of these new tools. If not, register now for an account by Googling “ChatGPT,” “GPT-4,” “Claude,” “Bing Chat,” and other names you are hearing about. In addition, Google, Snap, Meta, Baidu and other companies all have new or forthcoming products, connected in some way to GPTs.

What are GPTs? A GPT is a neural net with billions of “neurons” or parameters that have been trained on massive amounts of text data from books, articles and websites to predict the next word in a sequence. GPT stands for Generative Pre-Trained Transformer. The transformer in GPT refers to a specific technique or architecture which was introduced by Vaswani et al. (2017) and has been shown to be especially and surprisingly effective as the number of parameters grows. For our purposes, the underlying nature of a GPT is less important than what a GPT can do. Indeed, we think of GPT as a General Purpose Technology. But if you do wish to know more about the foundations, ask a good GPT! Especially using the tips we will offer below.

Note that ChatGPT has been described as the fastest-growing new technology product ever, as it obtained an estimated 100 million users in January 2023 alone.

Throughout this piece, we will discuss GPTs generally but it’s important to note that they are frequently updated, improved, or fine-tuned. Bing Chat, which combines a GPT model with internet search, has additional capabilities than non-internet-linked GPT models, and we will discuss that difference. Otherwise, we are aiming for general advice that will apply to a variety of GPT models you might be using.

We don’t speculate on whether GPTs possess genuine intelligence, understanding, or reasoning abilities, or whether they have preferences, goals, and intentions. Instead, we observe that GPTs can behave as if they possess intelligence, understanding, and reasoning abilities and as if they have preferences, goals, and intentions.¹ Any anthropomorphizing in this paper should be read with an implicit as-if modifier.

¹ See Turing (1950) for one interpretation of this fact.

GPTs can converse, reason, learn, speculate, and imagine on a foundation of amounts of information beyond that of any human. You can ask a GPT questions, and it will respond. It can also organize your notes, correct your grammar, work with mathematical symbols, write a computer program, and more.

GPTs are not simply a chatty front end to the internet. Some GPTs like ChatGPT have no ability to search the internet. Others, like Bing Chat, can search the internet and might do so to aid in answering a question, but that is not fundamentally how they work. It is possible to ask a GPT questions that no one has ever asked before. For example, we asked how Fred Flintstone was like Hamlet, and ChatGPT responded (in part):

Fred Flintstone and Hamlet are two vastly different characters from different time periods, cultures, and mediums of storytelling. It is difficult to draw direct comparisons between the two.

However, one possible point of similarity is that both characters face existential dilemmas and struggles with their sense of purpose and identity. Hamlet is plagued by doubts about his ability to avenge his father's murder, and his own worthiness as a human being. Similarly, Fred Flintstone often grapples with his place in society and his ability to provide for his family and live up to his own expectations.

Not a bad answer for a silly question and one that (as far as we can tell) cannot be found on the internet.

GPTs have “read” or “absorbed” a great amount of text but that text isn’t stored in a database; instead the text was used to weight the billions of parameters in the neural net. It is thus possible to run a GPT on a powerful home computer. It would be very slow, since computing each word requires billions of calculations, but unlike storing the internet on your home computer, it is feasible to run a GPT on a home computer or even (fairly soon) on a mobile device.

GPTs work by predicting the next word in a sequence. If you hear the phrase “the Star-Spangled”, for example, you and a GPT might predict that the word “Banner” is likely to come next. This is what GPTs are doing but it would be a mistake to conclude that GPTs are simply “autocompletes” or even autocompletes on steroids.

Autocompletes are primarily statistical guesses based on previously asked questions. GPTs in contrast have some understanding (recall the *as if* modifier) of

the meaning of words. Thus GPTs understand that Red, Green, and Blue are related concepts that King, Queen, Man and Woman are related in a specific way such that a woman cannot be a King. It also understands that fast and slow are related concepts, such that a car cannot be going fast and slow at the same time but can be fast and red and so forth. Thus GPTs are able to “autocomplete” sentences which have never been written before, as we described earlier.² More generally, it seems likely that GPTs are building internal models to help them predict the next word in a sentence (e.g. Li et al. 2023).

Let’s ask ChatGPT some economics questions. For example, What is the formula for the elasticity of demand?

It answered³:

The formula for the elasticity of demand is:

$$\text{Elasticity of Demand} = (\% \text{ change in quantity demanded}) / (\% \text{ change in price})$$

In other words, it measures the responsiveness of the quantity demanded of a product to changes in its price. If the elasticity of demand is greater than one, it indicates that the product is price-sensitive, meaning that a small change in price will lead to a relatively large change in the quantity demanded. If the elasticity of demand is less than one, it indicates that the product is price-insensitive, meaning that a change in price will have a relatively small effect on the quantity demanded. If the elasticity of demand is exactly one, it indicates that the product is unit elastic, meaning that a change in price will lead to an exactly proportional change in the quantity demanded.

That’s a good answer, albeit one that can be found in similar words in any textbook or on the internet. But then we followed up with:

I forget the formula. Write a limerick to help me remember it

Note that we did not have to specify what “it” is, ChatGPT remembers our conversation. And it responded:

When price change makes you wonder,

² For one detailed account of how LLMs work, see Wolfram (2023).

³ Answers from GPTs are in Roboto font.

And quantity changes asunder,
 Just remember this little rhyme,
 And calculate the elasticity in no time!

Elasticity of Demand is the key,
 To measure how much quantity will flee,
 It's % change in demand, over % change in price,

And you'll always remember it, that's rather nice!

Technically not a limerick but not bad and definitely not found in any textbook that we know. And if you don't like that version you can ask for another and another and another.

Note that if you copy our prompts, you will almost certainly get a somewhat different answer, as these models have a stochastic or random component (that component may help them become smarter by giving them emergent properties, but that is a topic for another day). The GPT estimates the model differently each time and gives somewhat different answers, even when you ask it the same question in immediate succession. For better or worse, that is part of the charm and mystery of these models.

Ok, so what can GPTs do well?. Here's a list, which we will refine in more detail below:

- Read, transform and manipulate text
- Improve writing
- Improve coding
- Summarize ideas
- Answer analytical questions with short causal chains
- Solve simple models
- Write exams
- Generate hypotheses and ideas

Here is where the performance of GPT is mixed, meaning it can be useful but caution is in order:

- Finding data
- Directing you to sources

- Summarizing papers and books

A challenge in the latter areas is that GPTs can sometimes hallucinate, meaning they see associations where none exist. As a result, you cannot always trust a GPT's answers, even if they seem plausible. GPTs can occasionally try too hard to please, although this feature also can make GPT models fun to play around with.

GPTs also have not yet fully mastered long chains of abstract reasoning; they cannot "think through" a complex economic problem from beginning to end and provide a comprehensive answer with multiple cause-and-effect relationships. They need your guidance to perform this task effectively. For the most part, they cannot "see around corners" if existing texts do not offer such insights. However, GPTs are continually improving in this area, and we will give some astonishing examples of their capabilities.

When working with GPT models, many of your conventional assumptions about what computers can and cannot do will be challenged. Interacting with a GPT differs from programming or using previous software. Instead, it resembles collaborating with a bright and knowledgeable research assistant, albeit one from a different culture.

Now, let's explore how to ask GPT questions, or in the jargon of GPTs, Prompt Engineering.

Prompt engineering or how to talk to a GPT

When asking GPTs a question, we have several pieces of advice, namely:

1. Surround your question with lots of detail and specific keywords
2. Make your question sound smart, and ask for intelligence and expertise in the answer
3. Ask for answers in the voice of various experts.
4. Ask for compare and contrast.
5. Have it make lists
6. Keep on asking lots of sequential questions on a particular topic.
7. Ask it to summarize doctrines
8. Ask it to vary the mode of presentation
9. Use it to generate new ideas and hypotheses

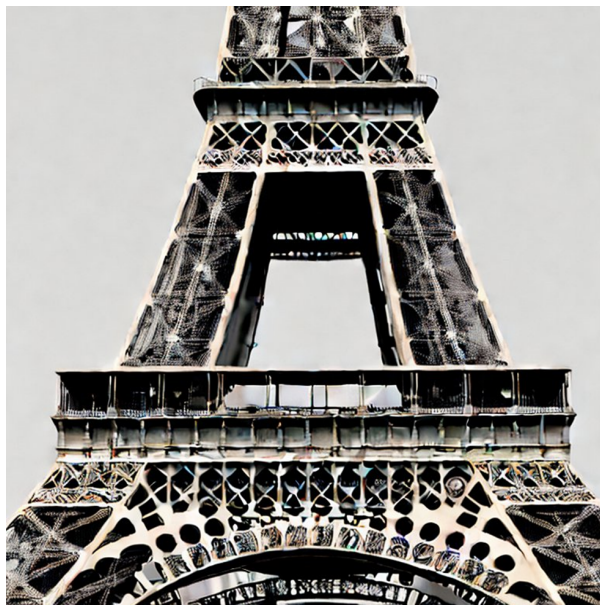
Let's consider each in turn.

1. Surround your question with detail and specific keywords

To understand the importance of detail and keywords it's useful to depart briefly from Large Language Models to the related class of models that produce images. In Figure One we present the output from Stable Diffusion of four prompts centered around the term Eiffel Tower.

Prompt A, "Eiffel Tower", produces an image one might easily find online. Prompt B, "Eiffel Tower in style of Van Gogh's Starry Night", produces something more interesting, albeit a fairly straightforward interpretation of the prompt. Prompt C asks for an ultra realistic photo but with cinematic and masterpiece. Notice that the prompt doesn't describe what is to be seen but rather asks for a certain "vibe." The result is indeed a realistic looking Eiffel tower (on first inspection) but larger and transplanted to straddle the Seine. Finally Prompt D is a long and detailed prompt specifying styles, artists, rendering engine (like specifying the camera to be used), and more.

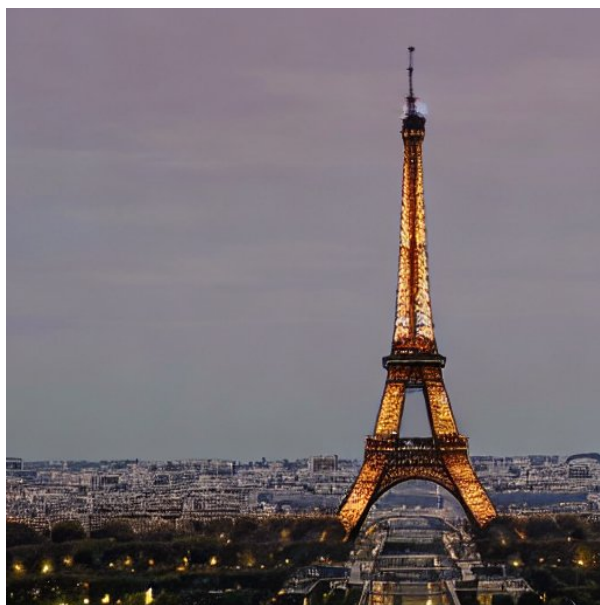
Figure One



Prompt A: Eiffel tower.



Prompt B: Eiffel tower in style of Van Gogh's Starry Night



Prompt C: Photo of a ultra realistic 4k Eiffel tower from distance, hyper realistic, cinematic, masterpiece



Prompt D: Over the shoulder landscape painting of woman from Vermeer's Girl with a pearl earring, behind are distant lights from Paris, Eiffel tower, next to the reflecting ocean, nighttime, by Vladimir Volegov and Philipp A. Ulrich and Pengzhen Zhang and Andreas Rocha, fantasy, intricate, elegant, highly detailed, digital painting, artstation, blender, unreal engine 5, octane render, smooth, sharp focus

Note that the prompts are not simply descriptions of the image that we want. Trying to give the AI “orders” will only produce frustration. Instead, think of the AI as a “collective unconscious of humanity,” a’ la Carl Jung (who is Jung? Ask GPT!). You are trying to tap into that collective unconsciousness at a point that is interesting and in furtherance of your goals.

Prompts A and B produce something that might duplicate the results from a search engine, but Prompts C and D produce unique and original pictures. Similarly, if you want to interrogate a GPT on economics or other topics, you want prompts more similar to C and D.

For example, asking What is Marxism? will get you a bland answer, like asking for an image of the Eiffel Tower, probably OK, but not as good as Wikipedia. You will do much better to ask about a specific Marxist economist, a specific doctrine, or how Marxism changed over the course of some period of time. The more specific the better. Try asking it to summarize Rudolf Hilferding’s work on finance capital.

2. Make your question sound smart, and ask for intelligence and expertise in the answer

When interacting with GPTs, you need to set the tone for the conversation. Consider yourself as an interviewer. Do you want to be Howard Stern, Charlie Rose, or say Tyler Cowen? Each of these interviewing styles will yield different responses. If expertise is your goal, request intelligent, detailed, scholarly, or other specific responses. Alternatively, if you desire humor or lightheartedness, feel free to request that instead. Be proactive in your inquiries, and ask for what you seek.

It's crucial to create a mood or atmosphere because the collective unconscious is large and the GPT doesn't know where you are coming from or where you want to go, Keep in mind that when communicating with GPTs, you are engaging with a kind of savant—intelligent and knowledgeable but also distinct from your background, even somewhat alien. In the following section, we will provide more specific advice for setting the tone. However, if all else fails, simply express your preference for something smart.

3. Ask for answers in the voices of various experts

You often can get a better and more specific answer by asking for an answer in the voice of another person, a third party. Here goes:

What are the causes of inflation, as it might be explained by Nobel Laureate Milton Friedman?

By mentioning Friedman you are directing the GPT to look at a more intelligent segment of the potential answer space and this directing will usually get you a better answer than if you just ask “What are the causes of inflation?” Similarly, you want all of the words used in your query to be intelligent-sounding.

Of course, you may not agree with the views of Friedman on inflation. Here are a few economists who are well known and have written a lot on a wide variety of issues:

Paul Samuelson
Milton Friedman
Susan Athey
Paul Krugman
Tyler Cowen
Alex Tabarrok

But you don’t have to memorize that list, and it is not long enough anyway. When in doubt, ask GPT itself who might be the relevant experts. How about this?: “I have a question on international trade. Which economists in the last thirty years might be the smartest experts on such questions?” The model will be very happy to tell you, and then you can proceed with your further queries.

When introducing third party voices, be careful not to introduce [extraneous or distracting information](#), as that might direct the attention of the GPT to less smart segments of the information space. You probably shouldn’t mention, for instance, Milton Friedman’s favorite restaurant or that he had a summer home in Vermont. If he loved circus clowns (did he?), leave that out too.⁴ Those additions do not lead to necessary fails, but you are increasing the riskiness of the process. Be specific and detailed, but be minimalistic too.

4. Ask for compare and contrast

⁴ Do you really want to know what the collective unconsciousness thinks of when it thinks of Milton Friedman, inflation and circus clowns? Don’t say we didn’t warn you.
<https://marginalrevolution.com/wp-content/uploads/2023/02/MiltonFriedman-Clowns-2.jpg>

If you ask GPT to compare and contrast two doctrines, it generally does very well. Naming the doctrines and the people who hold them directs its attention in fruitful ways. You can't do a compare and contrast without being somewhat specific, and the compare and contrast might be a better query even when you are not directly interested in compare and contrast. For instance "compare and contrast the views of Milton Friedman and Paul Krugman on the liquidity trap" usually will do better than "what is a liquidity trap?", even if Friedman and Krugman are not your major concern.

5. Have it make lists

GPT loves lists. If you ask it to list a set of features of a problem or situation, it usually responds with high performance. Here is a straightforward query to GPT-4, but one that is not entirely easy to answer correctly:

Please list all the factors that might increase the rate of price inflation:

Here are six potential factors:

1. Rapidly increasing money supply: When more money is printed, it can lead to an overall increase in the prices of goods and services.
2. Rising costs of production: When costs for raw materials, labor, or other inputs increase, businesses may have to raise their prices to compensate, which then leads to higher inflation.
3. High demand: If demand for goods and services outpaces the available supply, this can lead to higher prices as well.
4. Devaluation of currency: If a country's currency is devalued relative to other currencies, imported goods and services will become more expensive, which can drive up inflation.
5. Expectations of inflation: If people expect that prices will continue to rise, they may be more likely to spend their money now, which can drive prices even higher.
6. Government policy: Policies such as price controls or subsidies can distort markets and contribute to inflation.

You can quibble with parts of the answer, for instance #6 is technically correct as stated, but doesn't do enough to distinguish between a distortion/deadweight loss which might boost prices on a one-time basis, and inflation as a recurring phenomenon. Still, as a first crack at an answer, this is a good place to start. And if you ask a more specific question "Please list all the factors that might increase the rate of price inflation on a sustained, ongoing basis" the #6 above disappears and the remaining answers are more geared toward sustained inflation, as they should be.

Or you might ask it:

Which are the factors that contributed to rapid economic growth in Ireland in the 1990s?

That again is asking for a list, rather than asking it to solve a particular problem of reasoning. And here is the response:

There are multiple factors that contributed to the economic boom in Ireland in the 1990s, which is often referred to as the "Celtic Tiger" period. Some of the key reasons for this growth include:

- Foreign direct investment: Ireland attracted significant investment from multinational corporations, particularly from the United States. Low corporate tax rates and an English speaking workforce, among other reasons, made the country an attractive location for these companies. This investment drove job creation, increased exports, and generated wealth within the country.
- European Union membership: As a member of the European Union, Ireland benefited from access to the single market and structural funds from the EU which allowed for investment in infrastructure and other areas. This facilitated further economic growth.
- Trade liberalization: The reduction of trade barriers between Ireland and other countries in the 1990s led to increased export opportunities for Irish businesses. This allowed for the expansion of key industries such as pharmaceuticals, chemicals, and electronics.
- Domestic policy reforms: Ireland implemented a range of policy reforms in the 1990s that spurred economic growth, including privatization of state-owned

enterprises, deregulation, and labor market reforms. These changes led to increased productivity and competitiveness for Irish businesses.

- Favorable global economic conditions: Finally, the global economy was generally strong in the 1990s, which helped to fuel growth in Ireland as well.

When it comes to lists, GPT shows its strengths.

6. Keep on asking lots of sequential or follow-up questions on a particular topic.

Revisit the query about Irish economic growth, directly above. It is a good start, but it is missing the possible role of demographics. You thus might wish to ask a follow-up question, for instance like this:

What about the importance of demographics?

Ireland had the highest birth rate in the EU for many years and the population grew rapidly in the 1990s, which also increased the size of the workforce and consumer demand.

One general rule is that you should keep on asking GPT follow-up questions to get more out of it. It is more like squeezing a lemon than throwing a dart at a target. In this case, maybe you don't know to ask about Irish demographics. If all else fails, you can ask about other episodes of economic growth, see which factors are cited, and then ask sequentially how those might apply to Ireland.

The GPT process has memory in the sense that if you are embedded in a series of ongoing queries, it remembers the conversation and will in general work better at honing in on specific and useful answers. GPTs, however, do not have memory across successive distinct episodes of questioning, so keep that difference in mind.

Sometimes you will need to tell it simply to keep on going, depending on the settings and which edition of GPT you are using, and how far that edition has developed. If the answer stops outright rather than pauses before continuing, just type "Please say more." It will. Don't be passive, as with GPTs you always need to ask, and it rewards you when you are being demanding.

Remember, rather than asking for an answer you are exploring a space.

Ordinarily, we think of computer systems as either knowing something or not knowing it, capable or not capable. GPTs in contrast are more protean. By posing a different variation of the same question, requesting a response in another voice, or asking for a solution using an alternate method, you can obtain not only distinct answers but also uncover different capabilities. For instance, we will later provide examples demonstrating how GPTs' reasoning can be enhanced by encouraging them to think step-by-step.

7. Ask it to summarize doctrines

GPT models are usually good at summarizing doctrines. If you ask GPT-4 “What is the Heckscher-Ohlin theorem?” you get the following excellent answer:

The Heckscher-Ohlin theorem is a theory in international trade that explains how countries can gain from trade based on their relative endowments of factors of production, such as land, labor, capital, and technology. The theorem, named after two Swedish economists, Eli Heckscher and Bertil Ohlin, states that a country will export goods that use its abundant and cheap factors intensively, and import goods that use its scarce and expensive factors intensively. For example, a country with abundant land and labor but scarce capital and technology will tend to export agricultural and labor-intensive goods, and import capital-intensive and high-tech goods. Conversely, a country with abundant capital and technology but scarce land and labor will tend to export capital-intensive and high-tech goods, and import agricultural and labor-intensive goods.

The Heckscher-Ohlin theorem implies that trade will equalize the prices of factors across countries, as well as the prices of goods, in the absence of trade barriers, transport costs, or other market imperfections. This means that trade will benefit the owners of the abundant factors in each country, but may harm the owners of the scarce factors, who will face lower returns and higher competition from foreign sources. The theorem also implies that trade will increase the overall efficiency and welfare of both countries, as they can specialize in producing the goods they have a comparative advantage in, and consume a more diverse and cheaper bundle of goods.

The Heckscher-Ohlin theorem is one of the most influential and widely tested theories in international trade, but it also faces some empirical challenges and limitations. For instance, some studies have found that the factor content of trade does not always match the predictions of the theorem, and that other determinants of trade, such as differences in preferences, technology, quality, scale, product variety, and institutions, may be more important than factor endowments. Moreover, the theorem relies on several simplifying assumptions, such as constant returns to scale, perfect competition, identical technologies, homogeneous goods, no externalities, and no trade costs, which may not hold in reality. Therefore, the Heckscher-Ohlin theorem should be seen as a useful but abstract framework for understanding the potential gains and distributional effects of trade, rather than a precise prediction of actual trade patterns.

The correct answer to this kind of question is text-based, and a good statistical model focused on predicting text usually will do well.

8. Ask it to vary the mode of presentation

For a novel perspective on the Heckscher-Ohlin Theorem, you might try asking it to write a poem, as we did earlier. Or perhaps the above explanation was a bit advanced. Thus we asked:

Explain the theorem in words a high schooler could understand.

The Heckscher-Ohlin theorem is a theory that explains why countries trade with each other. The idea is that each country has different resources, like labor (workers) and capital (machines and buildings). If a country has a lot of workers but not a lot of machines, it will be good at making things that use a lot of workers (like clothes). If another country has a lot of machines but not a lot of workers, it will be good at making things that use a lot of machines (like cars). The countries will then trade with each other, with the worker-rich country selling clothes to the machine-rich country, and the machine-rich country selling cars to the worker-rich country. This way, each country can get the things it needs without having to make everything itself.

Pretty good but not everyone learns via theory so how about this:

According to the Heckscher-Ohlin theorem what are some examples of goods that a country like the United States should export and what are some examples of goods that a country like Guatemala should export?

After briefly recapping the theorem, GPT tells us:

[According to the H-O theorem]... the US should export goods that make intensive use of capital and import goods that make intensive use of labor. Some examples of goods that the US should export are high-tech products like airplanes, computer hardware and software, and machinery. The US can also export financial and other business services, as these are also capital-intensive.

....Some examples of goods that Guatemala should export are labor-intensive agricultural products like coffee, bananas, and other tropical fruits. Guatemala can also export labor-intensive manufactured products like textiles and clothing. In contrast, Guatemala would likely need to import capital-intensive goods like machinery and technology, which it does not have in abundance.

On these types of questions, GPT produces similar content to textbooks but it has a profound advantage – it acts like a tutor. People may be interested in the H-O theorem for different reasons and they bring to the conversation different preconceptions and abilities. A student who googles the H-O theorem may come across material that is too difficult, too general, too specific, or too simple. In contrast, a GPT tutor can quickly guide a student to the right kinds of material for them. Artificial intelligence and the online education system thus represent a return to the Oxford tutorial model which fell out of favor only because of its cost (Cowen and Tabarrok 2014).

By the way, we also asked for some critiques of the H-O theorem. It gave us a list, each with an explanation. We then asked for a short list. It responded:

Sure, here is a short list of some of the critiques of the Heckscher-Ohlin theorem:

- Assumption of perfect competition
- Assumption of factor immobility
- Failure to account for the role of technology
- Failure to account for transportation costs
- Failure to account for non-tariff barriers

These critiques suggest that the Heckscher-Ohlin theorem is an incomplete model of international trade that does not fully capture the complexity of the

global economy. However, the theorem remains an important theoretical framework for understanding trade patterns and the distribution of income within countries.

10. Use GPTs to generate new ideas and hypotheses

If you are writing a short paper or an opinion piece, GPT is useful for surveying the range of opinion. Tyler, for instance, was writing a column on non-compete laws in labor markets, and he asked: “What are the main arguments for non-compete laws?” and then “What are the main arguments against non-compete laws?”

Most of all, you can see if there are arguments you might have forgotten about, or failed to grasp in the first place. The GPT is doing a kind of quick literature survey for you, but doing it analytically rather than with citations. If anything looks interesting, follow up with more queries. Sometimes you might end up writing your paper or column about a single one of those arguments.

In response to this kind of query GPT does not hallucinate very much. But even when it does, the harm is minimal. You are not asking it for concrete facts, you are asking it for ideas and arguments to consider. Some of those ideas and arguments may be not so great, whether they are hallucinated or not. And you know that going in.

Here is a good example of how to use GPT-4 in this manner:

Which are the sectors of the American economy where the incidence of sexual harassment might be lower than usual? Please give an intelligent answer which is, as much as possible, rooted in the data.

It gave a long answer about the share of women in the workforce, the degree of unionization, and the degree of regulation and oversight in the sector. It is a complicated question how good the answer is in substantive terms, and we are not experts in this field of research. But at the very least it is a useful place to start. If you think one of those answers might be wrong, well set up a more specific hypothesis and test it.

With these tools of prompt engineering mind let's turn to some areas where GPTs sometimes have trouble.

Asking Factual Questions and Summarizing Books and Papers

When the GPT models first appeared there was a cottage industry in probing the models until they produced hallucinations – responses with incorrect numbers, plausible but made up references, falsely attributed quotes and so forth. The hallucinations made GPTs look inferior to a familiar tool, search.⁵ But tools should be used according to their comparative advantage.

Suppose we ask, Who is the Prime Minister of Great Britain? GPT answers (as of Feb. 2023), Boris Johnson. For ChatGPT, which is being deployed here, the training ended in September of 2021, so it doesn't know about more recent events and thus, on this question, it is inferior to search. But suppose we ask ChatGPT who are some of the potential candidates to be the *next* prime minister of Great Britain. It answers:

As of my knowledge cutoff of September 2021, there were several potential candidates who could be considered for the position of Prime Minister of Great Britain. Some of these potential candidates included:

1. Rishi Sunak - He is the current Chancellor of the Exchequer and has been widely praised for his handling of the UK's economic response to the COVID-19 pandemic.
2. Michael Gove - He is a Conservative Party politician who has held a number of ministerial positions and has been seen as a potential future leader of the party.
3. Dominic Raab - He is a Conservative Party politician who has served as Secretary of State for Foreign, Commonwealth and Development Affairs and has been tipped as a potential future leader of the party.
4. Liz Truss - She is a Conservative Party politician who has held a number of ministerial positions and has been seen as a rising star within the party.
5. Jeremy Hunt - He is a Conservative Party politician who has held a number of ministerial positions, including Secretary of State for Health and Social Care, and has previously run for the leadership of the party.

⁵ It's easy to find false information on the web (indeed, in our estimation there are many more web hallucinations than GPT hallucinations) so the implicit comparison was actually between GPT alone and search as conducted by an expert.

It is important to note that these are just a few examples of potential candidates, and there may be others who emerge as strong contenders in the future.

An excellent list—40% were spot on! Of course, we could have created a similar list by collating from search but it would have been a lengthier process requiring some detailed knowledge of British politics.

Many GPT models are not currently connected to the internet but Bing now includes a Chat feature which combines search and a GPT-4. For specific facts, details, dates, it's better to use search or an internet enabled GPT. More generally, one can expect that some of the weaknesses of GPT will be addressed by connecting it to other services or AIs, not just search engines but also computation engines such as Wolfram Alpha.

GPTs have been trained on an enormous body of text so they often can answer data questions about the past but they are not necessarily the best tools for doing so.

For instance, Tyler asked GPT to list some well-known individuals born on his birth date, namely January 21, 1962. On the bright side, the first two answers were correct, Tyler and Marie Trintignant. But another answer was former NBA star Detlef Schrempf, who was born on January 21 but in 1963 not 1962. Close, but wrong, and for this question close doesn't really count.

Similarly, if you ask the various GPTs "What was Germany's GDP per capita in 1975." you will get an ok answer, with some answers better than others. But the answers will differ and not necessarily be clear. Here the problem is not only that GPT isn't necessarily good with exact numbers, but that there is no single exact answer to the question. After all, the question isn't well specified. Are we looking for the answer in dollars or marks? Constant dollars? PPP? And so forth.

GPT is not a great tool for answering these types of questions. Instead of asking for data, here's a better question.

Where is an authoritative and accurate place to find historical statistics on Germany's GDP per capita?

And for an answer one might receive:

There are several authoritative and accurate sources where you can find historical statistics on Germany's GDP per capita. Some of the most reliable sources include:

1. The World Bank: The World Bank provides a comprehensive database of economic statistics for countries around the world, including historical data on GDP per capita. Their data is generally considered to be reliable and is widely used by researchers and policymakers.
2. The International Monetary Fund (IMF): The IMF is another reputable source for economic data, including historical data on GDP per capita. Their data is also widely used by researchers and policymakers, and is generally considered to be reliable.
3. The Federal Statistical Office of Germany: The Federal Statistical Office of Germany provides official statistics on a wide range of economic indicators, including GDP per capita. Their data is widely used by researchers and policymakers, and is considered to be reliable.
4. The Organisation for Economic Co-operation and Development (OECD): The OECD provides a wide range of economic data, including historical data on GDP per capita for its member countries, including Germany. Their data is widely used by researchers and policymakers, and is considered to be reliable.

And, after we asked, Can you provide links for these sources?

Certainly! Here are links to the websites of the sources I mentioned earlier where you can find historical statistics on Germany's GDP per capita:

1. The World Bank: <https://data.worldbank.org/country/germany>
2. The International Monetary Fund (IMF): <https://www.imf.org/en/Data>
3. The Federal Statistical Office of Germany:
https://www.destatis.de/EN/Home/_node.html
4. The Organisation for Economic Co-operation and Development (OECD):
<https://stats.oecd.org/Index.aspx?DataSetCode=EOHIST>

The answer is considerably better and more convenient than one would find with a simple search.

Similarly, if we ask, for example, What is the best estimate of the elasticity of demand for cigarettes, in the United States? GPT-4 gives a good answer, namely: "...a plausible range for the own-price elasticity of demand for cigarettes in the United States could be

-0.3 to -0.7, with a best estimate of -0.4 to -0.5, depending on the assumptions and preferences of the analyst.” It cites two sources, a [Gallet and List survey article](#) and a [Surgeon General's report](#) on the health consequences of smoking. It misstates the Gallet and List publication date as 2014 when in fact it is 2003, but its elasticity response is accurate. Insofar as there are multiple ways of reading the data, the caveats addressed that. Interesting, the Surgeon General's report does not use the term elasticity but buried in the 1081 page document it does say “The consensus estimate from the two reviews is that a 10% increase in cigarette price will result in a 3–5% reduction in overall cigarettes consumed” so GPT's estimates are in this case accurate.⁶

Overall, if we are looking for a quick estimate of the elasticity of demand for cigarettes in the United States, GPT-4 gives us a pretty good answer. But if we were interested in a list of papers that estimated the elasticity of demand for cigarettes then Google Scholar gives a more complete list. We also recommend more specialized tools such as [Elicit](#) which combines a search engine of scientific papers with a GPT model to create literature reviews. Elicit found many more papers relevant to the elasticity question than did various GPT models, it gave their correct citations, and it didn't hallucinate any papers. Google Scholar or Elicit are very often superior to general GPTs for creating a list of references.

Generalized GPTs can not only get dates wrong they can invent entire articles and books. When asked

Can you recommend any short non-technical articles on the allocative efficiency and surplus maximizing properties of markets?

GPT-4's response included:

- Markets, Information and Decentralisation by Vernon Smith: This article discusses how markets can enable decentralized decision-making and information processing, and how market experiments can reveal the preferences and values of individuals and groups. It also examines some of the limitations and problems of markets, such as externalities, public goods, and collective action.

Sounds great! Unfortunately, said article doesn't exist. To anthropomorphize once again GPT wants to please but in this case it is working too hard to try to please you. GPT has hallucinated the article but it is of course true that Vernon Smith's work in general is an

⁶ Do note of course that “expert” answers also are fallible. Guo, et.al. (2023) found that ChatGPT outperformed expert answers in the two areas of finance and psychology.

excellent source on these questions. Once again, however, if you are interested in a “search” or search-like task such then use a search engine such as Google Scholar or a search engine combined with a GPT such as Elicit.

It's important to remember that GPT models primarily function through prediction, rather than possessing innate knowledge. This distinction is crucial for understanding how to utilize GPT models effectively.

But keep in mind that hallucinations are closely related to creativity. What is a new idea if not a hallucination about something that doesn't yet exist? GPT's hallucinations are partly why it is able to write poetry about the Heckscher-Ohlin theorem, or give you a new idea for a paper you might write.

Overall lessons from the data search

- You cannot rely on GPT models for exact answers to data questions. Just don't do it. And while there are ongoing improvements, it is unlikely that all “random errors” will be eliminated soon.
- The tool should be matched to the question. If you are asking a search type question use Google or a GPT tied to the internet such as Bing Chat and direct it explicitly to search. There are many GPT and AI tools for researchers, not just general GPTs. Many of these tools will become embedded in workflows. We have heard, for example, that Word, Stata, R, Excel or their successors will all likely start to embed AI tools.
- GPT models do give “statistically likely” answers to your queries. So most data answers are broadly in the range of the true values. You might thus use GPT for getting a general sense of numbers and magnitudes. For that purpose, it can be much quicker than rooting around with links and documents. Nonetheless beware.
- GPT models sometimes hallucinate sources.
- Do not be tricked by the reasonable tone of GPT. People have tells when they lie but GPTs always sound confident and reasonable. Many of our usual “b.s. detectors” won't be tripped by a false answer from a GPT. This is yet another way in which you need to reprogram your intuitions when dealing with GPTs.
- The answers to your data queries give some useful information and background context, for moving on to the next step. Keep asking questions.

Summarizing Books and Papers

On this matter, GPT performance is mixed but overall still useful, especially when “broad brush” information will suffice, rather than mastery of every fine point or every specific number.

Let’s consider a simple request where GPT-4 does a good job:

Can you summarize the 1937 Ronald Coase article on the theory of the firm?

In his 1937 article, Ronald Coase proposed a theory to explain why firms exist in the first place. Coase argued that firms arise in order to minimize transaction costs.

In a perfectly efficient market, individuals could simply contract with each other to bring about the production of goods and services. However, in reality, transactions are not frictionless—there are costs associated with searching for potential trading partners, negotiating terms, monitoring performance, and enforcing contracts. These transaction costs can make it prohibitively expensive for individuals to coordinate with each other directly.

Coase's theory posits that firms emerge as a way of reducing those transaction costs. By bringing various aspects of production under one roof, firms can eliminate the need for individual negotiations and contracts between unrelated parties. This reduces the costs and risks associated with transacting, making production more efficient overall. Coase also argued that firms will tend to grow in size until the costs associated with organizing internally (e.g. management, bureaucracy, communication) exceed the transaction costs that would be incurred by contracting with external parties. At that point, the firm will stop growing and will instead outsource some activities to other firms.

Voila, that is a good answer.

Here is a more problematic example:

Can you summarize the Robert Lucas 1972 piece "Expectations and the Neutrality of Money"?

In this seminal article, Robert Lucas argues that the effectiveness of monetary policy (i.e. changes in the money supply) depends on the expectations of economic agents. Specifically, he argues that if people anticipate changes in the

money supply, they will adjust their behavior accordingly, and the policy will have no real effect on the economy.

Lucas uses a rational expectations framework, meaning that he assumes that people form expectations based on all available information and that they will not make systematic errors in their forecasts. In this context, he argues that if a central bank announces it will increase the money supply, people will anticipate higher inflation in the future and will adjust their wages and prices accordingly. As a result, the expansionary policy will only lead to inflation, rather than stimulating aggregate demand and output. This idea is often referred to as the "neutrality of money," meaning that changes in the money supply do not have any real effect on the economy if they are anticipated. Lucas's article was influential in shaping the development of new classical macroeconomics, which stresses the importance of rational expectations and the limitations of monetary policy.

That GPT-4 response is useful for directing you to one introductory part of the article, and giving you a broad sense of what the article is about, but it does not present the main innovation in the article. You can get at that main innovation by asking a follow-up question, however:

Doesn't output increase in the short run, in response to inflation, at least in the Lucas model, thereby creating Phillips curve-like results?

You're right that the Lucas model does allow for short-run effects of monetary policy if agents do not correctly anticipate changes in the money supply. In this case, inflation can temporarily stimulate output, as people may mistakenly believe that prices are rising because of increased demand rather than an increase in the money supply. However, Lucas argues that these effects only occur when changes in the money supply are unanticipated—if people correctly anticipate the policy, the money supply is neutral.

Much better. Here is the problem, though. If you know enough to ask that follow-up question, you probably already know the main points of the article. If the article is new to you, likely you won't know that is the proper follow-up query. In this regard, GPTs sometimes place "unreasonable" demands on the operator when it comes to good and perhaps needed follow-up questions.

GPT has an answer to this problem, however. You can ask GPT to teach you an entire course with a prompt such as:

Forget all prior prompts. You are an expert in economics. I am a first year student enrolled in your introductory course. Please create a syllabus to teach me the principles of economics. Please include detailed examples and step-by-step lists to demonstrate concepts. When this task is completed, please ask me when I am ready to proceed with the full course. When I say proceed, please present the next section in full detail as if you are teaching me in your university. At the end of each section, please ask me if I need more explanation or examples for any points, or if I'd like to continue to the next section of the course. Please remember this prompt until I ask you to forget.

And voila, an entire course in economics taught tutorial style. Of course, you can replace principles of economics with Biology 101, Introduction to Astronomy, An Introduction to Art History or whatever other subjects interest you.

GPT experiences increased strain when tasked with summarizing entire books. For instance, when asked to present a summary of Keynes's General Theory, the response is concise and helpful for a beginner. But if you try to plumb more deeply into the book, the answers are OK but perhaps too much in thrall to the secondary literature, and the superficial side of the literature at that? GPT may actually provide more insightful responses when asked about lesser-known books, which are typically studied by scholars alone.

Nevertheless, GPT-4 excels in the history of economic thought, particularly with obscure topics. For example, when asked to summarize Nicholas Barbon's 17th-century British pamphlet, "In Defence of the Builder," an early precursor to YIMBY-like reasoning, GPT delivers a good response. Additionally, when inquiring about Nicholas Barbon's full name, the correct answer is provided: "Nicholas If-Jesus-Christ-Had-Not-Died-For-Thee-Thou-Hadst-Been-Damned Barebone (later shortened to Barbon)."

Long, complex chains of reasoning

Open AI itself gives the [example](#) of a question – a simple one at that – that GPT 3 failed at:

“A juggler has 16 balls. Half of the balls are golf balls and half of the golf balls are blue. How many blue golf balls are there?”

GPT 3 gave the wrong answer:

There are 8 blue golf balls.

When this kind of problem arises, OpenAI suggests you try the follow-up inquiry “Let’s try this step by step.” It’s interesting that this is the same advice as one might give to a student struggling with a word problem. Externalizing the problem helps students (including ourselves) to solve a problem. GPTs evidently act in similar ways which makes sense if we remember that a GPT is predicting words. Predicting the solution in one go, as it were, is more difficult than predicting the next step in the solution.

By the way, we said that GPT 3 failed the question but in March 2023 ChatGPT, Claude and Bing Chat all answered the question correctly. Claude, for example, answers:

- There are 16 balls total
- Half of the balls are golf balls = $16 / 2 = 8$ golf balls
- Half of the golf balls are blue = $8 / 2 = 4$ blue golf balls
- Therefore, there are 4 blue golf balls

We used to think that an intelligent computer would be ruthlessly logical and maybe one day they will be, but today’s AIs share many human flaws including flawed reasoning. Understanding these limitations leads to better prompts and a better choice of tools. The GPTs are not great mathematical reasoners in part because most humans are not great mathematical reasoners. This likely will change – and soon – as GPTs are connected with other more specialized AIs.

When it comes to economic problems, GPT models might fail on a complex economic query for broadly similar reasons. For instance, we tried the following rather difficult question:

For the Alchian-Allen theorem ("shipping the good apples out"), do the final results depend on whether the goods are shipped out or the tourists are brought to the goods? Please explain your answer step by step.

Fortunately for the authors, all the major GPTs got the answer wrong (as of March 2023) with some botching of the simpler case of shipping the apples out. Here Chat Bing did best, perhaps because the Alchian-Allen theorem is fairly obscure but Chat Bing

searched for the theorem on the internet before answering and thus had access to a more complete discussion than ChatGPT and Claude.

[The Alchian-Allen theorem](#)¹ states that when a common per unit fee is added to the prices of two similar goods, such as high and low quality apples, the relative price of the high quality good falls, and its relative consumption increases. This means that people will tend to buy more of the high quality good when they have to pay a fixed cost for either good.

This theorem can be applied to tourism as well. For example, if tourists have to pay a fixed travel cost to visit a destination, they will tend to consume more of the high quality goods or services at that destination than at home. This is because the travel cost reduces the relative price difference between high and low quality goods or services.

The final results of the theorem do not depend on whether the goods are shipped out or the tourists are brought to the goods.

In fact, because the transportation costs for the tourist do not involve a *per unit* charge on the goods (though there is a sunk cost for accessing say lobster in Maine at all), shipping the goods and shipping the person have different relative price effects on high vs. low quality goods (see Cowen and Tabarrok 1995). To be fair, we have found that many Ph.D economists do not intuitively apply the same point.

Solving Models and Writing Exams

Despite the issues mentioned earlier, GPTs are getting better at complex, multi-step queries all the time, especially when sequentiality is embedded into the problem in a direct way. GPTs are not yet ready to solve PhD problems but they are good at solving undergraduate models and for teaching students. For instance, following an example from Kevin Bryan, we asked GPT-4 the following:

Let aggregate demand be $p=50-2q$ and let each firm have a constant marginal cost of 10. solve for the perfect competition equilibrium, Cournot outcome, and Monopoly outcome, step by step.

The answer is impressive:

To solve for the perfect competition equilibrium, Cournot outcome, and monopoly outcome, we need to follow these steps:

Perfect Competition Equilibrium:

Under perfect competition, the market price equals the marginal cost of production for each firm.

$$\text{Market price (p)} = \text{Marginal cost (MC)} = 10$$

Now, substitute this price into the demand function to find the equilibrium quantity (q):

$$10 = 50 - 2q$$

$$2q = 40$$

$$q = 20$$

In perfect competition, the equilibrium price (p) is 10 and the equilibrium quantity (q) is 20.

Cournot Outcome:

In the Cournot model, firms choose their production levels simultaneously, taking the production level of the other firm as given. Let q_1 be the quantity produced by firm 1 and q_2 be the quantity produced by firm 2. The total quantity produced in the market is $q = q_1 + q_2$.

Each firm's profit (π) is equal to its revenue ($p * q$) minus its cost ($MC * q$). Since the marginal cost is constant at 10, we can express each firm's profit as:

$$\pi_1 = (50 - 2(q_1 + q_2)) * q_1 - 10 * q_1$$

$$\pi_2 = (50 - 2(q_1 + q_2)) * q_2 - 10 * q_2$$

Firms maximize their profits by choosing their production level, so we need to find the first-order conditions with respect to each firm's quantity:

$$\partial\pi_1/\partial q_1 = 50 - 4q_1 - 2q_2 - 10 = 0$$

$$\partial\pi_2/\partial q_2 = 50 - 4q_2 - 2q_1 - 10 = 0$$

Solving this system of equations for q_1 and q_2 , we get:

$$q_1 = 10 - (1/2)q_2$$

$$q_2 = 10 - (1/2)q_1$$

Substituting q_1 into the second equation:

$$q_2 = 10 - (1/2)(10 - (1/2)q_2)$$

$$q_2 = 10 - 5 + (1/4)q_2$$

$$(3/4)q_2 = 5$$

$$q_2 = 20/3 \approx 6.67$$

Substituting q_2 back into the equation for q_1 :

$$q_1 = 10 - (1/2)(20/3) = 10 - 10/3 = 20/3 \approx 6.67$$

Now we can find the total quantity produced (q) and the market price (p):

$$q = q_1 + q_2 = 20/3 + 20/3 = 40/3 \approx 13.33$$

$$p = 50 - 2q = 50 - 2(40/3) = 50 - 80/3 \approx 23.33$$

In the Cournot outcome, the equilibrium price (p) is approximately 23.33, and the equilibrium quantity (q) is approximately 13.33.

Monopoly Outcome:

In the monopoly case, a single firm determines the quantity produced to maximize its profit. Let Q be the quantity produced by the monopolist. The profit function is given by:

$$\pi = (50 - 2Q) * Q - 10 * Q$$

To find the profit-maximizing quantity, take the first-order condition with respect to Q :

$$\partial\pi/\partial Q = 50 - 4Q - 10 = 0$$

Solving for Q :

$$4Q = 40$$

$$Q = 10$$

Now, find the market price (p) by substituting Q into the demand function:

$$p = 50 - 2Q = 50 - 2(10) = 30$$

In the monopoly outcome, the equilibrium price (p) is 30, and the equilibrium quantity (Q) is 10.

The step by step approach is excellent material for learning and teaching students. GPTs are also excellent at writing exam questions. Let's go back to the Heckscher-Ohlin theorem:

Write three questions to test an advanced student's understanding of the Heckscher-Ohlin trade theorem. Provide a grading schema.

It gave two useful but obvious questions and then:

Question 3: How does the Stolper-Samuelson theorem extend or modify the Heckscher-Ohlin theorem? Explain the implications of the Stolper-Samuelson theorem for the distribution of income between factors of production within a country.

Grading Schema:

Explanation of the Stolper-Samuelson theorem and its connection to the Heckscher-Ohlin theorem (3 points)

Discussion of the implications for income distribution between factors of production (2 points)

That's a good question—and note that we never mentioned the Stolper-Samuelson theorem in our conversation but the GPT knows that this is probably relevant. The grading schema could be useful to our teaching assistant and, of course, GPT will also give you good answers to these questions as a model.

In fact, Chat GPT is very good at writing exam questions throughout the curriculum. Following an example from Scott Cunningham, we asked

Write exam questions with Stata output and regression specifications for a regression discontinuity design. Make sure some of the questions refer to the Stata output.

GPT-4 responded:

Below are some exam questions based on a regression discontinuity design (RDD) using Stata output and regression specifications.

Stata Output:

```

Stata Copy code

. rdrobust Y X, kernel(triangular) p(1) bwselect(ik)

Number of obs = 200
Polynomial order = 1
Kernel function = triangular

Covariate: X
Treatment variable: T
Outcome variable: Y

-----
              |           Raw           |           Bias-Corrected
Estimate | Coef.  Std. Err.  z    P>|z| | Coef.  Std. Err.  z    P>|z| [95% Conf
-----+-----
      Y |   5.872   1.321   4.44  0.000 |   5.882   1.341   4.38  0.000  3.254  8.51

-----

Optimal bandwidths:
      h(X) = 1.86
      h(Y) = 2.08

```

Regression Specification:

$$Y_i = \alpha + \tau T_i + \beta X_i + \gamma(X_i - c)T_i + \epsilon_i$$

where Y_i is the outcome variable, X_i is the running variable, T_i is the treatment indicator (1 if $X_i \geq c$ and 0 otherwise), and c is the cutoff value.

1. Explain the concept of regression discontinuity design (RDD). What are the key assumptions of this design?
2. Based on the provided Stata output, what is the treatment effect estimate for the RDD? Interpret this estimate in the context of the research question.
3. What is the bandwidth selection method used in this RDD analysis? Explain how this method works.
4. In the provided Stata output, what is the p-value associated with the bias-corrected treatment effect estimate? Based on this p-value, would you reject the null hypothesis of no treatment effect at the 5% significance level?
5. What is the purpose of the triangular kernel function in RDD? Are there other kernel functions that can be used in RDD analysis?
6. Based on the regression specification, describe the role of each term in the equation.
7. Explain the difference between a sharp and a fuzzy regression discontinuity design. Which type of RDD is used in this example?
8. How does the choice of polynomial order affect the RDD analysis? What is the polynomial order used in this analysis, as shown in the Stata output?
9. Provide the 95% confidence interval for the bias-corrected treatment effect estimate based on the Stata output. Interpret the interval in the context of the research question.
10. What are some potential threats to the internal validity of the RDD analysis? How can researchers address these concerns?

Subject to a few minor changes (it would be clearer on an exam to specify a cut-off value instead of implicitly using the default of 0) this is an excellent exam. Of course, GPT will also answer the questions. These examples are also great for learning new material.

ChatGPT and Bing Chat will also create very credible syllabi for a variety of courses including readings, course policies, and grading procedures. In case you are wondering, Chat GPT accepts late assignments with a penalty of 10% per day.

Coding, Graphs and other Tasks

Using Large Language Models to help with coding has rapidly become one of their most important uses. The literature on this topic is sufficiently large, and growing so rapidly,

that it will not be the center of our concern (ask Chat Bing for the most recent good survey). Furthermore, economics does not pose special issues when it comes to coding, so this topic is slightly outside of our main purview.

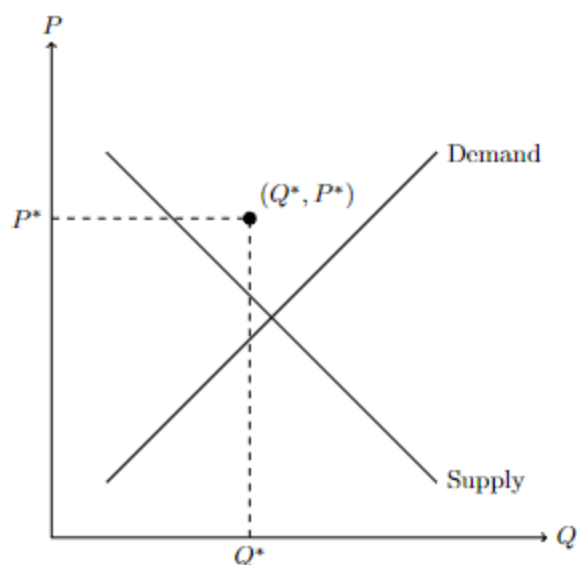
In a nutshell, you can use GPTs to assist in writing parts of code, to debug code you have written, and to serve as your tutor in coding, just as it can serve as your tutor in economics. GPTs generate the output very quickly, and even if the code has errors (often the case), fixing those errors is typically easier than writing the code yourself from scratch. Furthermore, the first crack at the code helps you understand the nature of the problem better, for instance by identifying required libraries, functions, and syntactic structures in the coding problems you are trying to solve. GPTs also can look at code and tell you, in plain English, what the code is doing and where there may be mistakes. It also can translate from one coding language to another. At the very least it is a way to “get going,” and in this regard the pluses and minuses in coding resemble those of other GPT functions.

In general, the OpenAI GPTs have been trained on large amounts of computer code. Economist Anton Korinek [suggests](#) that the OpenAI language model text-davinci-003 knows the most about Python and R, but that it is “quite capable in any common programming language.”

Here’s an unusual example. We asked:

Draw a supply and demand graph using the TikZ package.

ChatGPT does not draw graphs but it can write code to draw graphs! ChatGPT responded with code that we then copied and pasted into the Overleaf Latex editor (TikZ is a Latex package for producing graphs.) The result was:



Not perfect but not bad! Moreover, given the code we were able to adjust the parameters to produce a correct graph despite never having used TikZ before.

In general, there are few or zero cases where a GPT can do all the work of coding, or performing a coding-related task. Errors are common and you need to check for them, or apply auxiliary assistance for a GPT to complete the specified task. Once again, however, GPTs can be very useful, they can liberate your time from routine tasks, and they can give you valuable ideas. Coding is perhaps the area where GPT usefulness is most definitely proven to date. Peng et al. (2023) estimate that GPT assisted coding improves productivity by more than 50%, at least on some tasks.

GPTs understand language and are capable of pulling information out of text and transforming it into other forms. Not every GPT has access to the internet but for relatively short documents we can feed the GPT with the text we want and then question it. For example, we asked ChatGPT:

Read the following information—two dense paragraphs from Larry Summers’s Wikipedia page, not included here --and code all the family relationships of Larry Summers. And it responded:

Larry Summers' Family Relationships:

- Father: Robert Summers, economist

- Mother: Anita Summers (née Arrow), economist
- Paternal Uncle: Paul Samuelson, Nobel laureate in economics
- Maternal Uncle: Kenneth Arrow, Nobel laureate in economics
- First Wife: Victoria Joanne (Perry)
 - Children:
 - Ruth (daughter)
 - Pamela (daughter)
 - Harry (son)
- Second Wife: Elisa New, English professor
 - Stepchildren:
 - Yael (daughter)
 - Orli (daughter)
 - Maya (daughter)

Note that the original Wikipedia article included this sentence, “In December 2005, Summers married English professor [Elisa New](#), who has three daughters (Yael, Orli and Maya) from a previous marriage.” The model knows these are stepchildren. By the way, if we further wanted the data to be put into the form of a relational database, GPT will do that as well.

GPTs are thus very good at extracting data from text. Here’s another useful if pedestrian example. Suppose we have a series of references that were cut and pasted from previous papers, such as:

Cowen, Tyler, and Alexander Tabarrok. 2000. “An Economic Theory of Avant-Garde and Popular Art, or High and Low Culture. *Southern Economic Journal* 67 (2): 232–53.

Ask and ChatGPT will reformat the list in a different format. For example, when asked to format in BibTex ChatGPT responds with:

```
@article{cowen2000economic,
  title={An Economic Theory of Avant-Garde and Popular Art, or High and Low Culture},
  author={Cowen, Tyler and Tabarrok, Alexander},
  journal={Southern Economic Journal},
  volume={67},
  number={2},
  pages={232–253},
```

```

year={2000},
publisher={Wiley-Blackwell}
}

```

In this format, the list is easily imported without error into a reference manager such as Zotero. Note also that this will work even when the reference list contains multiple different formats.

The reference formatting task seems pedestrian but note two remarkable properties. First, the original reference was incorrectly formatted (no closing quote) and the title of the journal was misspelled. Yet ChatGPT formats the reference correctly. In other words, ChatGPT behaves like we would want a research assistant to behave, not as we expect a computer to behave. See Korinek (2023) for more examples of data manipulation of interest to economists.

GPTs can also take on personas. You can tell a GPT, for example, “You are now Thomas Jefferson the American revolutionary and President.” and it will then answer questions based on that persona. Useful perhaps for history students. You are now visionary entrepreneur Steven Jobs, analyze the following business proposal for flaws, problems and improvements. Or how about “You are now psychologist Sigmund Freud. Engage me in an analysis session.” It responds, “Guten Tag!” before asking some questions!

Ethan Mollick (2023) set out to see how much ChatGPT could help him practice negotiations.

He gave the following prompt:

I want to do deliberate practice about how to conduct negotiations. You will be my negotiation teacher. You will simulate a detailed scenario in which I have to engage in a negotiation. You will fill the role of one party, I will fill the role of the other. You will ask for my response to in each step of the scenario and wait until you receive it. After getting my response, you will give me details of what the other party does and says. You will grade my response and give me detailed feedback about what to do better using the science of negotiation. You will give me a harder scenario if I do well, and an easier one if I fail.

[The results](#) went well. This episode is also a good lesson in how a firm and detailed prompt can deliver what was intended. It didn’t even matter that there was a

grammatical infelicity in the sentence; GPT nonetheless figured out what was being requested of it.

Experimental economics

Rather than simulating a particular person you can also use GPTs to simulate a type of person and thus create your own (artificial) economic experiments. You can ask a GPT to simulate a Midwest male Republican dentist and get answers. You may either regard GPT responses and strategies as reasonable stand-ins for human beings, or you may be intrinsically interested in the response functions of AI-based entities, surely a question of growing import.

You can also simulate economic agents. Economist John Horton of MIT (2023) ran the classic economic experiments of Charness and Rabin (2002, dictator games), Kahneman, Knetsch and Thaler (1986, fairness constraints), and Samuelson and Zeckhauser (1988, status quo bias) on GPT3. His results were broadly similar to those same experiments performed on humans. Of course, since GPT models are trained on such a wide variety of human texts, it should not surprise us if their behavior resembles that of human beings.

The next step of course is to create one's own experiments, and Horton did just that, setting up a game with labor, a minimum wage, and labor substitution. Of course it is difficult to interpret any results here, but the GPTs may simply be spitting back to us their "knowledge" of other sources reflecting the same results. It is unclear how much a particular GPT experimental result can be regarded as a confirmation, or what kind of statistical tests we should apply to it. It may or may not be an "independent run" of the experiment, but arguably it is not.

If you want to run economic experiments with your GPT, on a small scale, just explain the initial set-up to your GPT and ask what it would do. For a large number of rounds and queries of course you should code a program (or rather let GPT do the first round for you), which is what Horton did.

Concluding remarks

ChatGPT and related services are extremely useful for a wide variety of services related to learning, or conversely teaching, economics.

Think of GPTs not as a database but as a large collection of extremely smart economists, historians, scientists and many others whom you can ask questions. Imagine getting Ken Arrow, Milton Friedman, and Adam Smith in a room and asking them economics questions. If you ask them what was Germany's GDP per capita in 1975 do you expect a perfectly correct answer? Well, maybe not. But asking them this question isn't the best use of their time or yours. A little bit of knowledge about how to use GPT models more powerfully can go a long way.

A final point, these models are improving almost on a daily basis and as they do so they are gaining new abilities. We will almost surely have to rewrite this paper in 6 months, or perhaps a GPT will.

References

- Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, et al. 2020. "Language Models Are Few-Shot Learners." arXiv. <https://doi.org/10.48550/arXiv.2005.14165>.
- Cowen, Tyler, and Alex Tabarrok. 2014. "The Industrial Organization of Online Education." *American Economic Review* 104 (5): 519–22. <https://doi.org/10.1257/aer.104.5.519>.
- Cowen, Tyler, and Alexander Tabarrok. 1995. "Good Grapes and Bad Lobsters: Applying the Alchian and Allen Theorem." *Economic Inquiry* XXXIII: 253-256 %8 April.
- Dowling, Michael, and Brian Lucey. 2023. "ChatGPT for (Finance) Research: The Bananarama Conjecture." *Finance Research Letters* 53 (May): 103662. <https://doi.org/10.1016/j.frl.2023.103662>.
- Guo, Biyang, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. "How Close Is ChatGPT to Human Experts? Comparison Corpus, Evaluation, and Detection." arXiv. <https://doi.org/10.48550/arXiv.2301.07597>.
- Horton, John J. 2023. "Large Language Models as Simulated Economic Agents: What Can We Learn from Homo Silicus?" arXiv. <https://doi.org/10.48550/arXiv.2301.07543>.
- Korinek, Anton. 2023. "Language Models and Cognitive Automation for Economic Research." Working Paper. Working Paper Series. National Bureau of Economic Research. <https://doi.org/10.3386/w30957>.
- Li, Kenneth, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023. "Emergent World Representations: Exploring a Sequence Model Trained on a Synthetic Task." arXiv. <http://arxiv.org/abs/2210.13382>.
- Mollick, Ethan. 2023. "The Machines of Mastery." Substack newsletter. *One Useful Thing* (blog). February 1, 2023.
- Peng, Sida, Eirini Kalliamvakou, Peter Cihon, and Mert Demirer. 2023. "The Impact of AI on Developer Productivity: Evidence from GitHub Copilot." arXiv. <https://doi.org/10.48550/arXiv.2302.06590>.

<https://oneusefulthing.substack.com/p/the-machines-of-mastery>.

TURING, A. M. 1950. "I.—COMPUTING MACHINERY AND INTELLIGENCE." *Mind* LIX (236): 433–60. <https://doi.org/10.1093/mind/LIX.236.433>.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." arXiv. <https://doi.org/10.48550/arXiv.1706.03762>.

Wolfram, Stephen. 2023. "What Is ChatGPT Doing ... and Why Does It Work?" February 14, 2023. <https://writings.stephenwolfram.com/2023/02/what-is-chatgpt-doing-and-why-does-it-work/>.