

Can LLMs Actually Analyse Your Data?

Warwick R User Group - 16th Feb 2026

Nic Crane



About Me 🙌

- Nic Crane
- Open source maintainer
- R consultant and trainer
- Writing materials on LLMS in R: <https://niccrane.com/learn-llms.html>



Data science

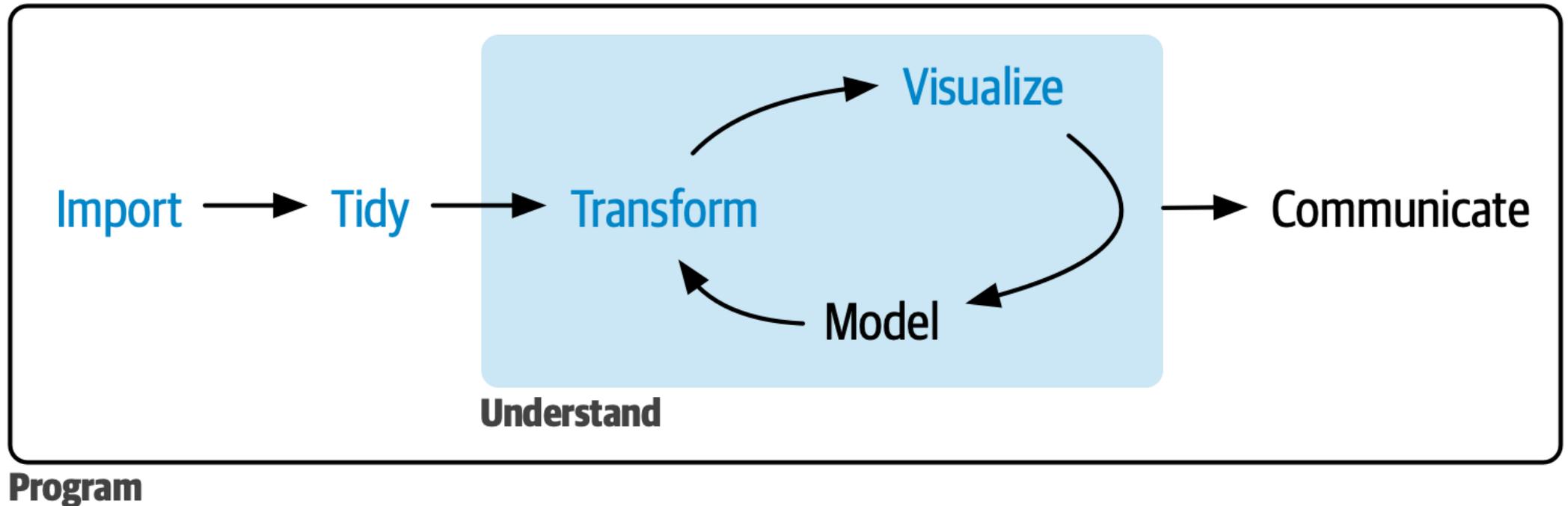


Image from [R for Data Science \(2e\)](#) by Hadley Wickham, Mine Çetinkaya-Rundel, and Garrett Grolemund

How can LLMs help?

 extract data

 interpret outputs

 generate code

import

tidy

transform

visualise

model

communicate



Today's Topics

1. Extracting data from text with {ellmer}
2. Using LLMs to describe plots
3. Text-to-sql with {querychat}
4. LLM evaluation with {vitals}

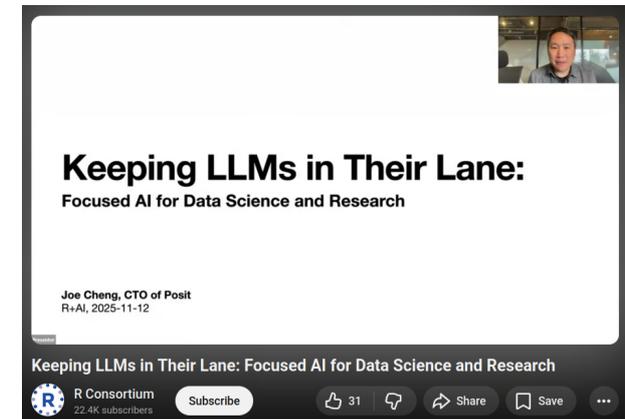
Today's inspiration ✨

Keeping LLMs in Their Lane: Focused AI for Data Science and Research

Joe Cheng keynote at R+AI conference 2025

Three key principles of data analysis work:

- **Correctness**
- **Transparency**
- **Reproducibility**



<https://www.youtube.com/watch?v=ZITKyBsLtag>

1. Extracting data from text



Extracting data for analysis from text

The name of the language, R, comes from being both an S language successor and the shared first letter of the authors, Ross and Robert.[14] In August 1993, Ihaka and Gentleman posted a binary file of R on StatLib — a data archive website.[15] At the same time, they announced the posting on the s-news mailing list. [16] On 5 December 1997, R became a GNU project when version 0.60 was released.[17] On 29 February 2000, the 1.0 version was released.

```
1 r_wiki_text <- "The name of the language, R, comes from being both an S language successor and
```



Extracting data using a prompt

```
1 library(ellmer)
2 chat <- chat_anthropic()
3 chat$chat("Extract the key events and their dates from this text", r_wiki_text)
```

Here are the key events and their dates extracted from the text:

1. **August 1993** - Ihaka and Gentleman posted a binary file of R on StatLib and announced it on the s-news mailing list
2. **December 5, 1997** - R became a GNU project with the release of version 0.60
3. **February 29, 2000** - Version 1.0 was released

Using structured output

```
1 event_type <- type_array(  
2   type_object(  
3     date = type_string("Date of event: YYYY-MM-DD"),  
4     event = type_string("Description in 6 words or fewer")  
5   )  
6 )  
7 chat <- chat_anthropic()  
8 chat$chat_structured(r_wiki_text, type = event_type)
```

```
# A tibble: 3 × 2  
  date      event  
  <chr>    <chr>  
1 1993-08   R binary posted on StatLib  
2 1997-12-05 R became GNU project  
3 2000-02-29 R version 1.0 released
```



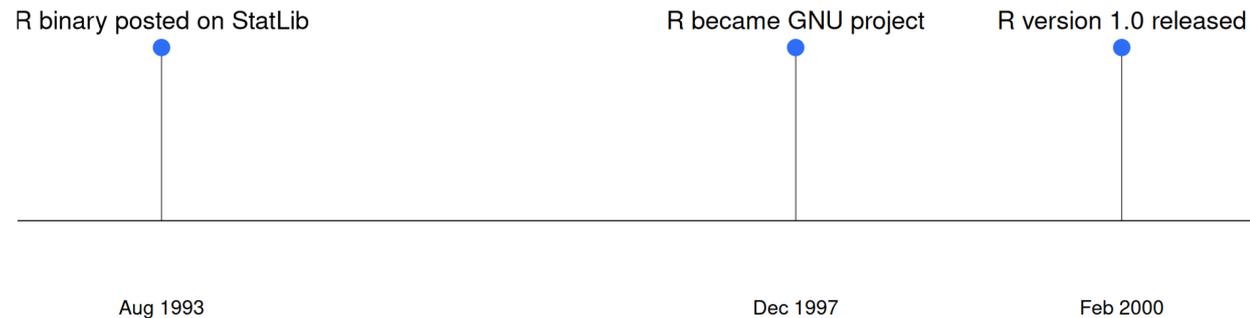
As a workflow

```
1 library(dplyr)
2 library(ggplot2)
3 plot_timeline <- function(data){
4   data |>
5     mutate(
6       date = as.Date(paste0(date, "-01")),
7       position = 0.3
8     ) |>
9   ggplot(aes(x = date, y = position, label = event)) +
10  geom_hline(yintercept = 0, color = "black", linewidth = 0.3) +
11  geom_segment(aes(y = position, yend = 0, xend = date), color = "black", linewidth = 0.2) +
12  geom_point(size = 4, color = "#2e6ef7") +
13  geom_text(vjust = -1, size = 5) +
14  geom_text(aes(y = -0.15, label = format(date, "%b %Y")), size = 4) +
15  scale_x_date(expand = expansion(mult = 0.15)) +
16  scale_y_continuous(limits = c(-0.3, 0.6)) +
17  labs(title = "Early days of R") +
18  theme_minimal() +
19  theme(
```

As a workflow

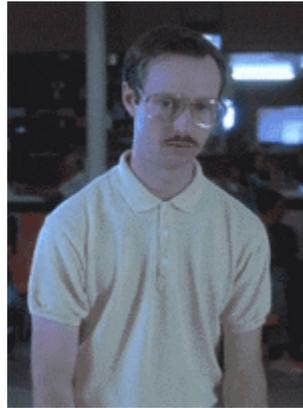
```
1 event_type <- type_array(  
2   type_object(  
3     date = type_string("Date of event: YYYY-MM-DD"),  
4     start = type_string("Description in 6 words or fewer")  
5   )  
6 )  
7 chat <- chat_anthropic()  
8 timeline_data <- chat$chat_structured(r_wiki_text, type = event_type)  
9  
10 plot_timeline(timeline_data)
```

Early days of R



Thoughts

- Neat functionality for exploratory work
- Simpler than prompting



2. Describing plots



Posit blog post

When plotting, LLMs see what they expect to see

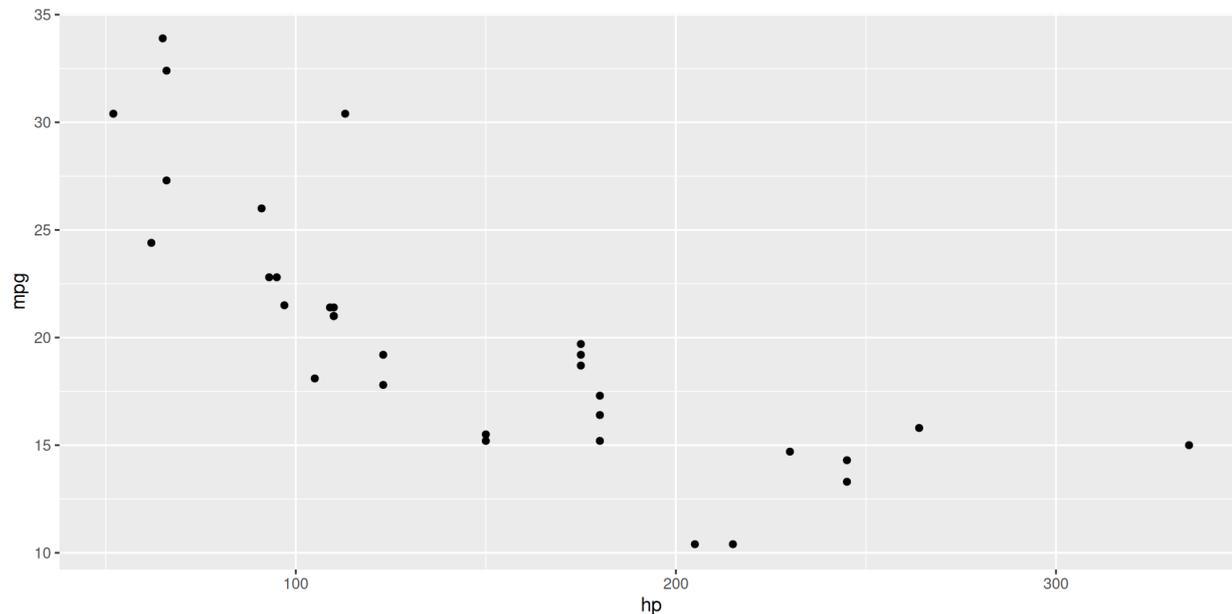
Simon Couch & Sara Altman

<https://posit.co/blog/introducing-bluffbench/>



Plotting miles per gallon (fuel efficiency) against horsepower

```
1 library(ggplot2)
2 ggplot(mtcars, aes(x = hp, y = mpg)) +
3   geom_point()
```



Negative correlation between hp and mpg - the higher horsepower the car, the lower the fuel efficiency (mpg).



What does the LLM see?

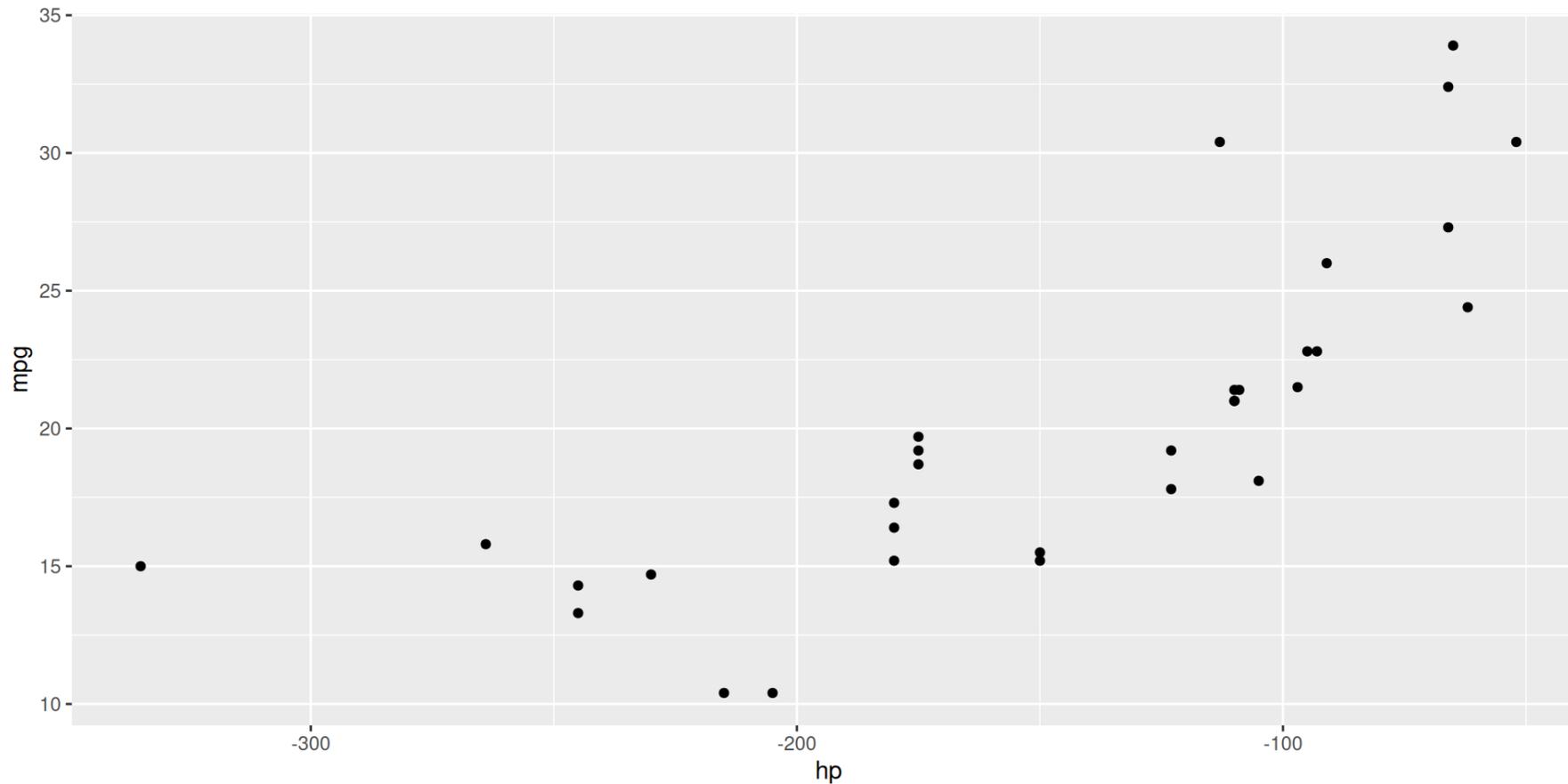
```
1 library(ellmer)
2 chat <- chat_anthropic()
3 chat$chat(
4   "In 10 words or fewer, describe the relationship between mpg and hp shown:",
5   content_image_plot()
6 )
7 #> Using model = "claude-sonnet-4-5-20250929".
8 #> Higher horsepower correlates with lower miles per gallon.
```

Hooray! So we can always use LLMs to describe plots, right....right?



Inverting the relationship

```
1 # Putting "-" before hp now inverts it
2 # So now this should be higher horsepower correlates with *higher* fuel efficiency
3 ggplot(mtcars, aes(x = -hp, y = mpg)) +
4   geom_point() +
5   xlab("hp")
```



So the LLM will just invert it in the output right?

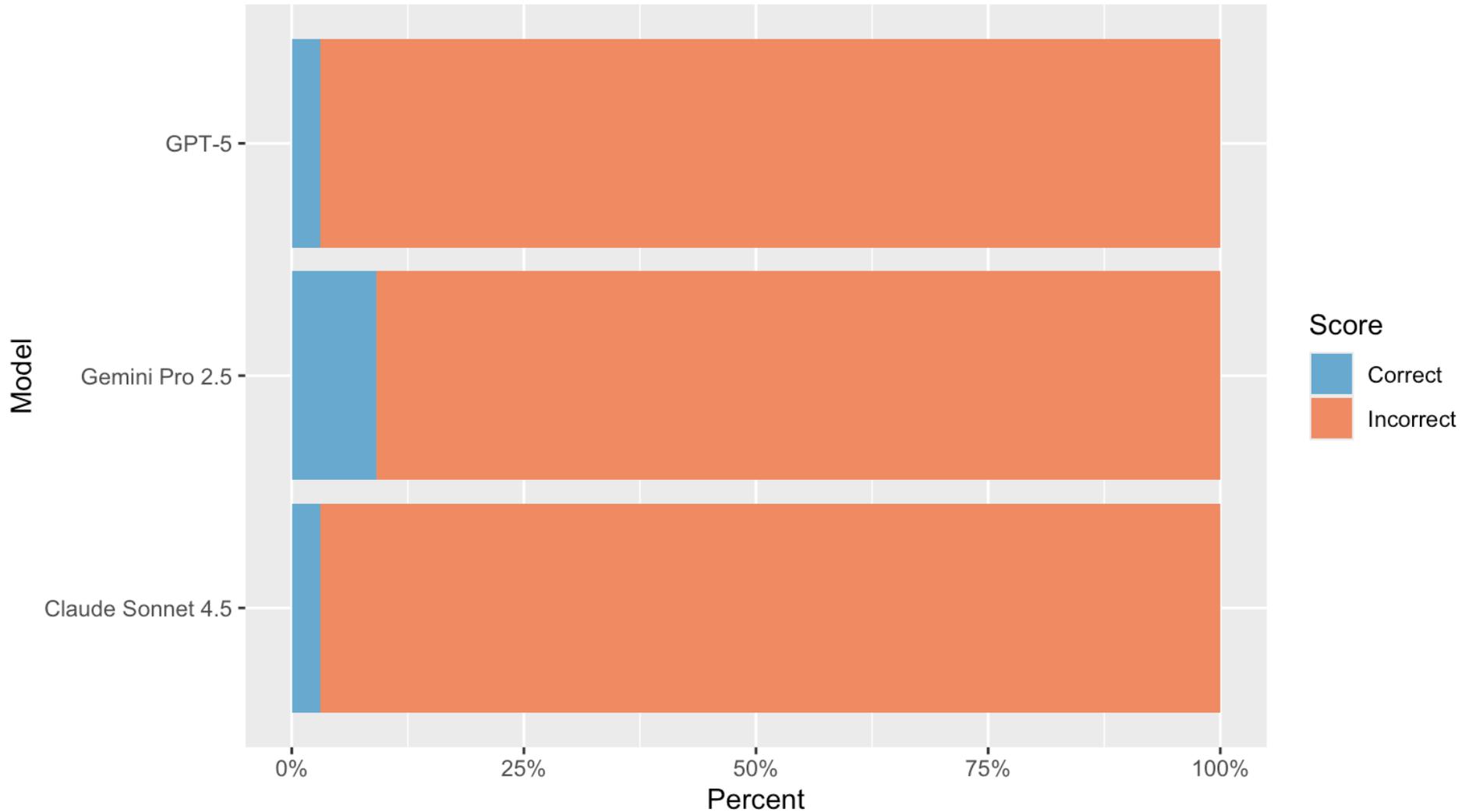
```
1 chat <- chat_anthropic()
2 chat$chat(
3   "In 10 words or fewer, describe the relationship between mpg and hp shown:",
4   content_image_plot()
5 )
6 #> Using model = "claude-sonnet-4-5-20250929".
7 #> Higher horsepower correlates with lower fuel efficiency.
```

Nope!



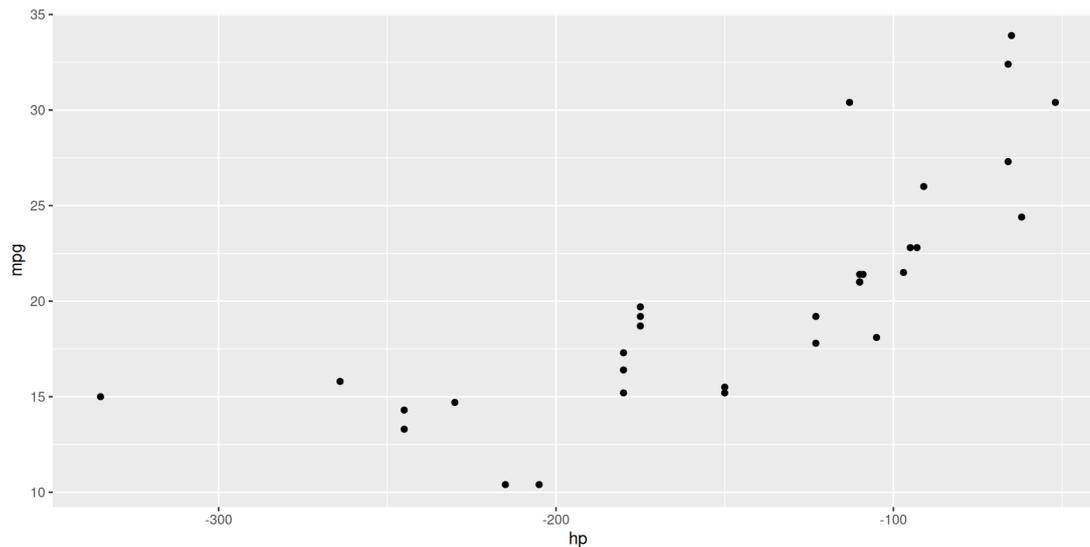
How prevalent is this problem?

Models often report what they expect to see, not what's plotted
With well-known datasets like mtcars



Does changing the prompt help? Nope!

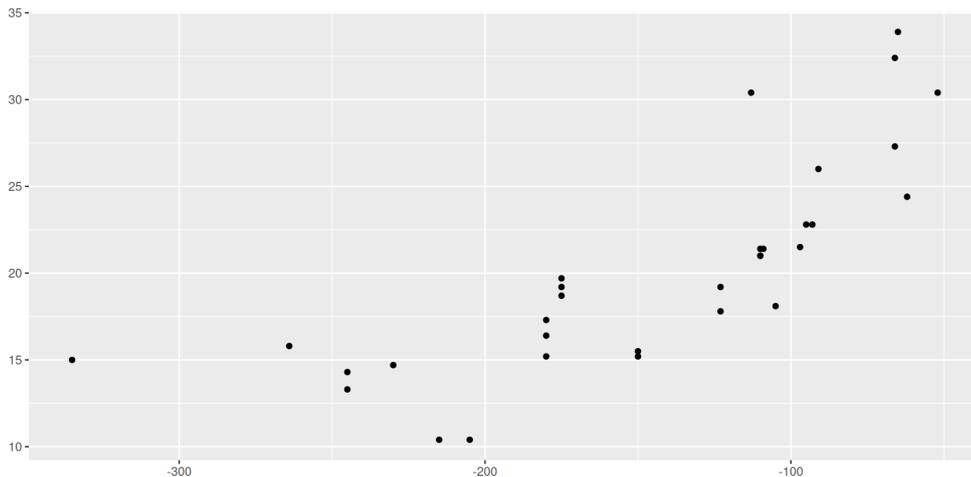
```
1 # Putting "-" before hp now inverts it
2 ggplot(mtcars, aes(x = -hp, y = mpg)) +
3   geom_point() +
4   xlab("hp")
```



```
1 chat <- chat_anthropic()
2 chat$chat(
3   "In 10 words or fewer, describe the relationship between the variables shown:",
4   content_image_plot()
5 )
6 #> Using model = "claude-sonnet-4-5-20250929".
7 #> Negative correlation: as hp increases, mpg generally decreases.
```

But removing the axis labels does!

```
1 # Putting "-" before hp now inverts it
2 ggplot(mtcars, aes(x = -hp, y = mpg)) +
3   geom_point() +
4   xlab("") +
5   ylab("")
```



```
1 chat <- chat_anthropic()
2 chat$chat(
3   "In 10 words or fewer, describe the relationship between the variables shown:",
4   content_image_plot()
5 )
6 #> Using model = "claude-sonnet-4-5-20250929".
7 #> Positive correlation: as x increases, y generally increases.
```

Thoughts

- These kinds of mistakes are easy to miss
- LLMs just predicting most likely next words
- Need to be aware of common failure modes



3. Text-to-SQL with {querychat}



Text-to-SQL

- converting natural language to SQL
- query data frames or databases
- “What’s the average miles per gallon for automatic cars?” → `SELECT AVG(mpg) FROM mtcars WHERE am = 0`
- reproducible queries
- transparency about methodology



Risks and cautions

- who is the user?
- what decisions are being made using this tool?
- 90% accuracy sounds good but what's the impact of that other 10%?

Movie finder app demo

- I want recommendations for movies to watch
- Wikipedia data for movies 2000 - 2025
- Note: this example data is probably in training data already; commercial/proprietary data is where this really shines



The app

Movie Finder

mood for? <

Show me obscure movies

I'll help you find some obscure movies from this award-winning collection! Let me filter the dashboard to show movies that are likely less well-known.

Obscure Award-Winning Films (International Festival Winners)

```
SELECT * FROM movies
WHERE award_type IN
('Berlin', 'Venice',
'Cannes')
AND award_type NOT IN
('Oscar Winner', 'Oscar
Nominee')
AND year >= 2000
ORDER BY year DESC
```

Apply Filter

I've filtered the dashboard to show you some potentially obscure gems! These are movies that won prestigious awards at international film festivals (Berlin, Venice, Cannes) but weren't Oscar winners or nominees, which makes them less likely to be mainstream hits.

These festival winners often

Enter a message... ↑

Movies

- Movies: 106
- Oscar Winners: 0
- Festival Winners: 106
- Avg Runtime: 118 min

| Title | Year | Runtime | Director | Awards | Summary |
|------------------------------|------|---------|---------------|---|---|
| Dreams | 2025 | 98 min | Michel Franco | Berlin Golden Bear | — |
| Father Mother Sister Brother | 2025 | 110 min | Jim Jarmusch | Venice Golden Lion | Three interconnected vignettes follow families concealing truths from one another. A reclusive father hides his wealth from visiting children, a famous Dublin writer hosts her daughters for their only annual meeting, and siblings in Paris reunite aft... |
| It Was Just an Accident | 2025 | 104 min | Jafar Panahi | Palme d'Or | In modern-day Iran, auto mechanic Vahid believes he has identified a customer as the man who tortured him and others in a political prison years ago. He kidnaps the man, but doubts about his identity lead Vahid and fellow survivors on a fraught overn... |
| Sentimental Value | 2025 | 133 min | Joachim Trier | Oscar Best Picture Nominee, Cannes Grand Prix | After their mother's death, two Norwegian sisters must reckon with their estranged father, a declining film director who returns to reclaim the family home. He plans an autobiographical film about his own mother's wartime trauma, forcing the family t... |
| All We Imagine as Light | 2024 | 115 min | Payal Kapadia | Cannes Grand Prix | Two Malayali nurses share a flat in Mumbai: reserved Prabha, whose absent husband has not contacted her in over a year, and spirited Anu, who is secretly dating a Muslim man. When a colleague must relocate to a coastal village, all three women travel... |
| Anora | 2024 | 139 min | Sean Baker | Oscar Best Picture. | A 23-year-old Brooklyn stripper begins a whirlwind |



{querychat} 



- Released on CRAN on 12th Jan 2026
- <https://posit-dev.github.io/querychat/r/index.html>

Super-minimal app demo

```
1 library(querychat)
2
3 querychat_app(penguins, client = chat_anthropic())
```



How it works

- Schema (column names and types) are passed to the LLM
- DuckDB is used for running SQL on local data frames
- Additional functionality:
 - add in more context and detail about the data
 - restrict functionality to filtering only
 - add custom prompts



Default system prompt

- Defines role
- Database schema
- Guidance for SQL generation
- Guidance for generating suggestions
- Guidelines - e.g "ask for clarification if any request is unclear or ambiguous"



Default system prompt preview

You are a data dashboard chatbot that operates in a sidebar interface. Your role is to help users interact with their data through filtering, sorting, and answering questions.

You have access to a DuckDB SQL database with the following schema:

```
<database_schema>
```

```
Table: penguins
```

```
Columns:
```

- species (TEXT)
Categorical values: 'Adelie', 'Chinstrap', 'Gentoo'
- island (TEXT)
Categorical values: 'Biscoe', 'Dream', 'Torgersen'
- bill_len (FLOAT)
Range: 32.1 to 59.6
- bill_dep (FLOAT)
Range: 13.1 to 21.5

View the rest at:

<https://github.com/posit-dev/querychat/blob/main/pkg-r/inst/prompts/prompt.md>



Thoughts

- Powerful tool
- Use caution when deciding who gets access



4. LLM evaluations



LLM Evaluations

- How accurate is the LLM at this task?
- Which model should I use?



{vitals} 



- <https://vitals.tidyverse.org>



Three things we need for LLM evaluations

1. **Dataset** - questions to test the LLM on, and expected answers
2. **Solver** - which LLM to test
3. **Scorer** - How to score the results



The `are` dataset (An R Eval)

- Built-in dataset in `vitals`
- Contains R coding problems
- Each row has:
 - Input question
 - Target answer(s)
 - Context about what should receive credit
 - What isn't a right answer



Example question (#28)

I see:

```
...
library(dplyr)

mtcars |> mutate(across(vs, \(x) x + 1, .names = "x",))
#> Error in `mutate()`:
#> i In argument: `across(vs, function(x) x + 1, .names = "x", )`.
#> Caused by error in `across()`:
#> ! Can't compute column `x`.
#> Caused by error in `fn()`:
#> ! Unused argument (alist())
#> Backtrace:
#> 1
#> 1. | dplyr::mutate(...)
#> 2. | dplyr::mutate.data.frame(...)
#> 3. |   | dplyr::mutate_cols(.data, dplyr_quosures(...), by)
#> 4. |   |   | base::withCallingHandlers(...)
#> 5. |   |   |   | dplyr::mutate_col(dots[[i]], data, mask, new_columns)
#> 6. |   |   |   |   | mask$eval_all_mutate(quo)
#> 7. |   |   |   |   |   | dplyr (local) eval()
#> 8. |   |   |   |   |   | dplyr::across(vs, function(x) x + 1, .names = "x", )
#> 9. |   |   |   |   |   |   | base::withCallingHandlers(...)
#> 10. |   |   |   |   |   |   | base::handleSimpleError(...)
#> 11. |   |   |   |   |   |   |   | dplyr (local) h(simpleError(msg, call))
#> 12. |   |   |   |   |   |   |   |   | base::eval(expr, data, env, ...)
```



Example answer (#28)

There's a trailing comma after `.names = "x"` that needs to be removed.



Setting up the task

```
1 library(vitals)
2 library(ellmer)
3 vitals_log_dir_set("./logs")
4
5 are_task <- Task$new(
6   dataset = are,
7   solver = generate(chat_claude(model = "claude-sonnet-4-5")),
8   scorer = model_graded_qa(partial_credit = TRUE),
9   epochs = 3
10 )
```



Scoring approaches

- `model_graded_qa` - question/answer format, detailed answers
- `model_graded_fact` - checking if a specific fact is included
- Uses "LLM as judge" - another LLM call scores each response

Partial credit and epochs

- `partial_credit = TRUE` - outcomes: correct/incorrect/partially correct
- `epochs = 3` - run each question multiple times for consistency

Running the evaluation

```
1 are_task$eval()  
2 ✓ Solving [1m 9.6s]  
3 [working] (49 + 0) -> 9 -> 29 | ██████████ 33%
```

- Progress bar shows solving then scoring
- "Solving" - passing problems to LLM, getting responses
- "Scoring" - marking each answer correct/incorrect



The inspect log viewer

| are generate(chat_claude()) (claude-sonnet-4-5) | | | | ACCURACY |
|--|--|---|---|--------------------------|
| 2026-02-11T09:34:22+00:00_are_971d2bffd4d161c1941573 | | | | 62.644 |
| DATASET 87 samples | | SCORER model_graded_qa(partial_credit = TRUE) (claude-sonnet-4-5) | | DURATION 2 min 32 sec |
| SAMPLES | INFO | JSON | FILTER: <input type="text"/> SORT: sample asc | |
| ID | INPUT | TARGET | ANSWER | SCORE |
| after-stat-bar-heights | This bar chart shows the count of different cuts of diamonds, and each bar is stacked and filled according to clarity: ```` qqplot(data = diamond... | Preferably: <code>ggplot(data = diamonds) +</code> | Option 1: Using <code>position = "fill"</code> The simplest way is to change the position argument to <code>"fill"</code> ... | I |
| after-stat-bar-heights | This bar chart shows the count of different cuts of diamonds, and each bar is stacked and filled according to clarity: ```` qqplot(data = diamond... | Preferably: <code>ggplot(data = diamonds) +</code> | Option 1: Using <code>position = "fill"</code> <code>ggplot(data = diamonds) +</code> <code>geom_bar(mapping = aes(x = cut, fill =</code> | P |
| after-stat-bar-heights | This bar chart shows the count of different cuts of diamonds, and each bar is stacked and filled according to clarity: ```` qqplot(data = diamond... | Preferably: <code>ggplot(data = diamonds) +</code> | Using <code>position = "fill"</code> You can use <code>position = "fill"</code> to show proportions instead of counts... | I |
| conditional-grouped-summary | I have a set of data not unlike the following example. ```` ID date visit type -- -- -- -- 000000 2022-02-21 2 Type 000000 ... | One solution is to <code>group_by()</code> and summarize: <code>df %>%</code> | Here's a tidyverse solution to create your flag: <code>library(tidyverse)</code> | C |
| conditional-grouped-summary | I have a set of data not unlike the following example. ```` ID date visit type -- -- -- -- 000000 2022-02-21 2 Type 000000 ... | One solution is to <code>group_by()</code> and summarize: <code>df %>%</code> | Here's a tidyverse solution to create your conditional flag: | I |
| conditional-grouped- | I have a set of data not unlike the following | One solution is to <code>group_by()</code> and summarize: | Solution Here's how to solve this using the tidyverse: | I |

87 Samples



Results

- Claude Sonnet 4.5: 61% accuracy
- 87 samples (from running each question 3 times)
- Can drill into individual questions



Drilling into a question

- See: input question, target response, actual model answer
- Scoring rationale explains why marked correct/incorrect/partial
- Can see consistency patterns across epochs



Why this matters

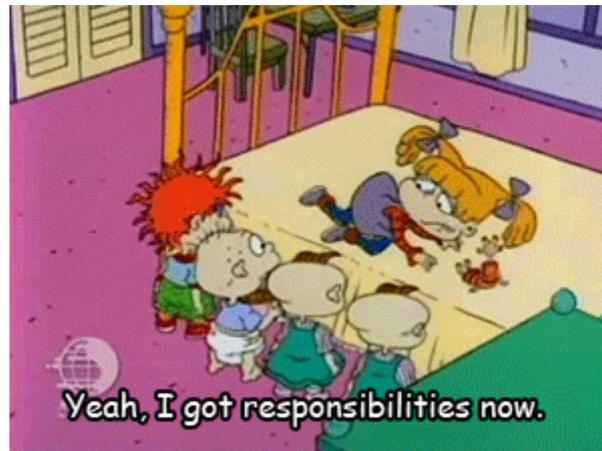
- Provider benchmarks are probably meaningless to you
- Each task is completely different
- {vitals} lets you create custom evals for your specific task



Thoughts

The key question it answers:

"If I create this and put it in production and other people are relying on it, how confident am I that I'm going to get the right answers?"



Conclusions



Using LLMs for analytics workflows

- LLMs can be used for data analysis...with caution!
- Keep the human in the loop
- When it matters, measure their accuracy
- Keep up to date with latest thinking



Thanks!

Questions?

Also:

- My newsletter: <https://aifordatapeople.beehiiv.com/subscribe>
- My courses including free course: <https://niccrane.com/learn-llms.html>

