From ChatGPT to AI Agents: Practical RNA-seq Workflows with Codex CLI

Qi Sun
Bioinformatics Facility
Cornell University

Use an AI agent to analyze RNA-seq data

Put all your data files into a folder

Open AI agent

A chatbot will open

Analysis protocol (guardrail)

You:	Load rnaseq.md
Agent:	File loaded. It is an RNA-seq pipeline.
You:	Write a script to analyze RNA-seq data in this directory.
Agent:	Created. Do you want me to execute it?

(You can exit “agent” at this point. If you need its help later, you can “wake up” the “agent”)

Run the script

If the run fails ...

You: It failed. Help me to fix it.

Agent: I'm checking the script and recent run artifacts to find the actual failure point before changing anything.

... the agent explores for a short time ...

Agent: The failure is early and specific: current nf-core/rnaseq no longer accepts --aligner star.

Agent: I'm patching the script now.

Agent: The fatal error is fixed. Re-run the same script run_rnaseq_nfcore.sh

What the AI agent can do:

"Identify Differentially expressed genes between conditions A and B"

"Make a PCA plot for all samples, 'A' samples in red, 'B' samples in blue. "

"Interpret this PCA plot for me."

"Write a method section for my manuscript"

Outline of this lecture

1. **What is an AI Agent?**

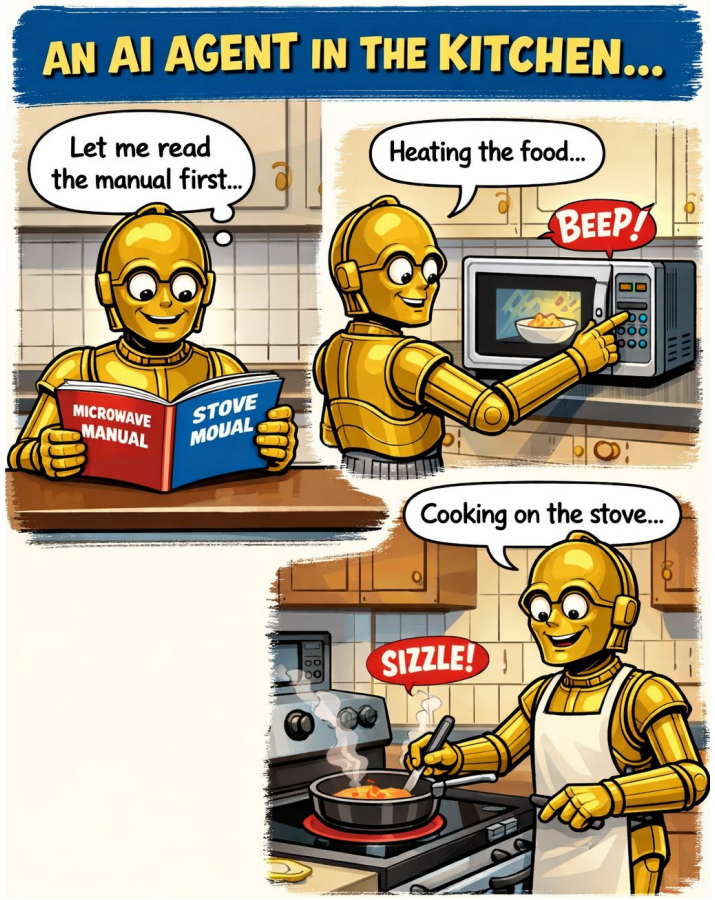
2. **How to work with an AI agent?**
   - **Guardrails**
     - Hardware/software constraints
     - RNA-seq analysis protocol
   - **Context Management**
     - Avoid context contamination
     - Manage context size (token limits)
     - RAG to bring in missing information

3. **Cost of using AI**
   - Commercial service providers: ChatGPT, Claude, Gemini, CoPilot
   - Free models and free tools

What is an AI Agent

An AI agent is like a humanoid robot in your kitchen. If you need to heat food, the robot operates appliances like a microwave or stove.

— Jensen Huang (NVIDIA)



Bioinformatics Data Analysis

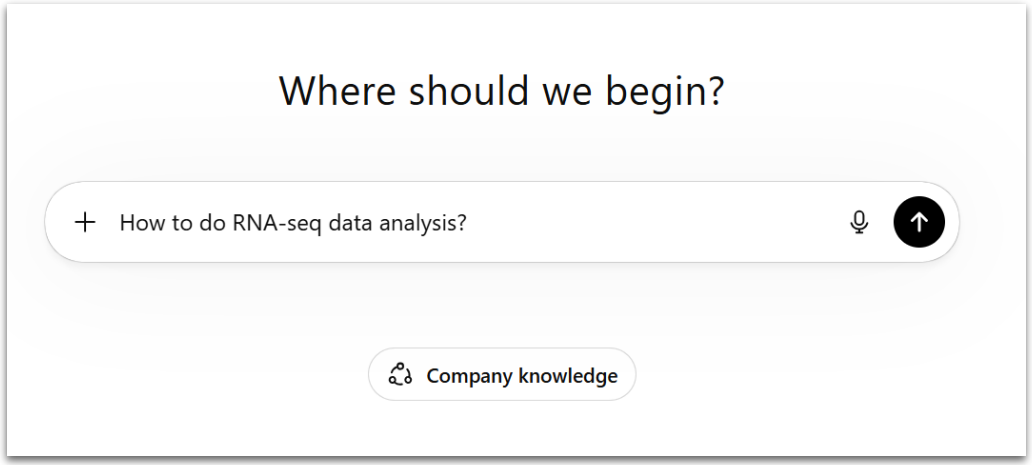
The chef ➡ You (Researcher)

The robot ➡ AI Agent (Codex CLI)

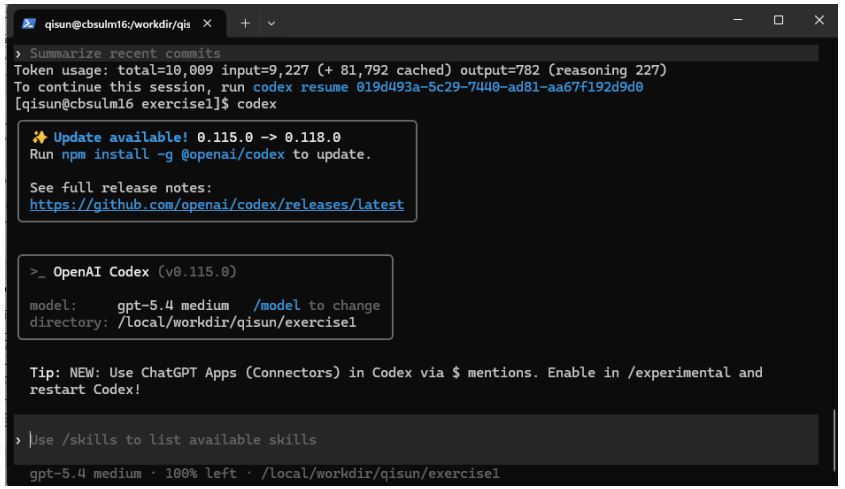
Microwave, stove, et al. ➡ Bioinformatics Tools (STAR, DESeq2, etc.)

ChatGPT vs Codex CLI

ChatGPT – OpenAI's Chat App



Codex CLI - OpenAI's Agent



Difference:

- Codex has access to your Linux environment and your files
- Codex can run software in the shell
- If things go wrong, Codex can troubleshoot

Working with an AI Agent - Guardrails

For example, for RNA-seq data analysis

Hardware level

- Where I will run my data analysis, local or a Slurm cluster?
- How many CPU cores I have access to?
- Which volume to use as scratch disk?

Software level

- What software to use?
- Some important parameters?

The facility will provide BioHPC system related guardrail file for you and help you to customize it.

- Use ChatGPT for help, or consult other people;
- Supply some reading materials to ChatGPT (**RAG**)

How to provide guardrails to the agent?

- **It is a software built in plain English language.**
- **This software enables Codex to do RNA-seq data analysis for you.**

Through prompts

You: Use nf-core/rnaseq for RNA-seq data analysis
Agent: … …
You: Use STAR as aligner.
Agent: … …
You: Uses sampleMeta.txt if available
Agent: … …
You: For qc, only do FastQC, MultiQC and Alignment stats. skip bigWig.
Agent: … …

Through a file

RNA-seq Pipeline
- nf-core/rnaseq
- use STAR as aligner
- Convert the sample into the right format for nf-core/rnaseq
- Create custom.config, use 8 CPU cores for the STAR aligner. Run multiple STAR jobs in parallel.
- Use all CPU cores of the server to run the pipeline
- For QC, only do FastQC, MultiQC and Alignment stats. Skip bigWig.
- Workflow: counts → PCA
- After read counts are generated, work with user to run DESeq2 to get differentially expressed genes.
- Switch to R 4.4.3 by running command "module load R/4.4.3". DEseq2 is installed in this version of R, no need to verify.

It is preferrable to use files than prompts.

- You have a record;
- Re-usable
- More efficient;

Two guardrail files will be used in this workshop

- **AGENTS.md (System prompt)**

Provided by Bioinformatics Facility. You can copy this file to your ~/.codex/ directory. This file provides general information about BioHPC, e.g. use /workdir as scratch disk, software installation in /programs directory, use docker1 command to run docker.

* CLAUDE.md for Claude

- **rnaseq.md (protocol, guardrail, skill file)**

Provided by Cornell Genomics Facility. We will collaborate with other BRC facilities to develop more data type specific guardrail files.

* All guardrail files will be kept in /programs/ai_pipelines

Customize BioHPC System Prompt

AGENTS.md for BioHPC

...
Installed Software Lookup- Before installing software, check this web page first: https://biohpc.cornell.edu/lab/userguide.aspx?a=software.

...

guardrail file lookup
- If not in current directory, find it under /programs/ai_pipelines/. Each analysis type has its own subdirectory.

...

- Use GNU parallel for job parallelization. It is installed and in default path.

...

-- Do not run a pipeline automatically, instead, put the command in a bash script

Use Software installed by BioHPC team?

Where to find guardrail files?

If you use a SLURM cluster, change this line.

Default behavior.

You might need to modify the guardrail files

rnaseq.md

RNA-seq Pipeline
- nf-core/rnaseq
- use STAR as aligner
- Convert the sample into the right format for nf-core/rnaseq
- Create custom.config, use 8 CPU cores for the STAR aligner. Run multiple STAR jobs in parallel.
- **Use all CPU cores of the server to run the pipeline**
- For QC, only do FastQC, MultiQC and Alignment stats. Skip bigWig.
- Workflow: counts → PCA
- After read counts are generated, work with user to run DESeq2 to get differentially expressed genes.
- Switch to R 4.4.3 by running command "module load R/4.4.3". DEseq2 is installed in this version of R, no need to verify.

How many CPU cores to use?

Skip some QCs in the pipeline?

Which R version to use?

(You can simply put “Use R version 4.4.3”, because the AGENT.md file specify the BioHPC website to find instructions to run software. More detailed information like this save agent time to look up the information.)

It takes several iterations to develop a workable guardrail

ChatGPT

Example prompts:

- What software should I use?
- Is there an existing pipeline, preferrable from nf-core?
- Can you make a guardrail for me?

Codex

- **Test the guardrail.** If you modify the guardrail file, or have multiple versions, make sure not to test them in the same chat session.
- **Fix the guardrail.**

If the run fails. Tell Codex:

Fix the guardrail for me.

AI has a tendency to agree with you. Explicitly tell ChatGPT to be more critical. In Codex, you might want to ask “flag any questionable assumptions”

File names

- **AGENTS.md * - System prompt**

  - Up to two files with this name. One in “~/.codex/”, and the other in project directory.
  - If there are conflicts, the one in project directory override the one in “~/codex/”

* Claude uses CLAUDE.md file.

- **Extra files - can be any names**

  - Load the files through prompt, e.g. “load rnaseq.md”

File format: Markdown.

(You can use a text editor to create or modify a markdown file)

AGENTS.md file for BioHPC

```
## working directory
-    for working directory, use directory under `/workdir/$USER`, not my home directory `/home/$USER`. If a software
     needs large temporary directory, do not use /tmp, create /workdir/$USER/tmp and use this as tmp directory.

## Server Environment Constraints
-    Do not use `docker` on this server.
-    Use `docker1` directly for all Docker operations on this server.
-    I was given permission to run sudo docker through `docker1`, which is a wrapper for "sudo docker".
-    For examples such as listing images, showing containers, building images, or running containers, use commands like:
    - `docker1 images`
    - `docker1 ps`

...

## Installed Software Lookup
- Before installing software, check this web page first: https://biohpc.cornell.edu/lab/userguide.aspx?a=software. In this
HTML page, all software names are listed, with hyperlink point to a page with instructions how to run this software.
```

Indentation matters

- The Markdown “tags” are not semantically required by Codex;
- But “tags” and indentation make it easier for human to read, and for AI to parse (indentation is meaningful) .

Three example guardrail files for RNA-seq analysis

Minimal

```
## RNA-seq Pipeline
- nf-core/rnaseq
```

More constraints added

```
## RNA-seq Pipeline
- Use nf-core/rnaseq
- use STAR as aligner
- Convert the sample into the right format for nf-core/rnaseq
- Create custom.config, use 8 CPU cores for the STAR aligner. Run multiple STAR jobs in parallel.
- Use all CPU cores of the server to run the pipeline
- For QC, only do FastQC, MultiQC and Alignment stats. Skip bigWig.
- Workflow: counts → PCA
- After read counts are generated, work with user to run DESeq2 to get differentially expressed genes.
- Switch to R 4.4.3 by running command "module load R/4.4.3". DEseq2 is installed in this version of R, no need to verify.
```

Detailed guardrail by genomics facility (partial)

```
## ----------------------------
## RNA-seq Pipeline
## ----------------------------
- version: 1.0
- use nf-core/rnaseq to process RNA-seq data, using the current nf-core/rnaseq S
TAR plus Salmon route.
- Includes trimming, automatic strandedness detection, Salmon-based quantificati
on after STAR alignment, and DESeq2 QC analysis. FastQC is skipped.
- Create custom.config, use 6 CPU cores for the STAR aligner. Run multiple STAR
jobs in parallel.

## ----------------------------
## INPUT SPECIFICATION
## ----------------------------
inputs:
 sample_sheet:
  required_columns:
   - sample
   - condition
   - fastq_1
   - fastq_2
  optional_columns:
   - batch
   - replicate

 fastq_requirements:
  pattern: "*_{R1,R2}.fastq.gz"
  paired_end: true
  gzip_required: true
...
```

Reproducibility and Verification

- Where guardrails specify behavior -> deterministic
- Where guardrails are silent -> probabilistic

Unit tests for verification

- Verify scientifically critical decisions
- Accept variations unspecified in guardrail

rnaseq.md

```
## RNA-seq Pipeline
- Use nf-core/rnaseq
- use STAR as aligner
- Convert the sample into the right format for nf-core/rnaseq
- Create custom.config, use 8 CPU cores for the STAR aligner. Run multiple STAR jobs in parallel.
- Use all CPU cores of the server to run the pipeline
- For QC, only do FastQC, MultiQC and Alignment stats. Skip bigWig.
- Workflow: counts → PCA
- After read counts are generated, work with user to run DESeq2 to get differentially expressed genes.
- Switch to R 4.4.3 by running command "module load R/4.4.3". DEseq2 is installed in this version of R, no need to verify.
```

Can you help to improve it?

A Guardrail Hackathon?

- Build BioAgent that can do different type of data analysis.
- Coding with plain English language (or other languages?)

Some ideas:

Make it a full functional BioAgent, that can interact with users, to define parameters, to do extra analysis, visualization.

Working with AI Agent - Context management

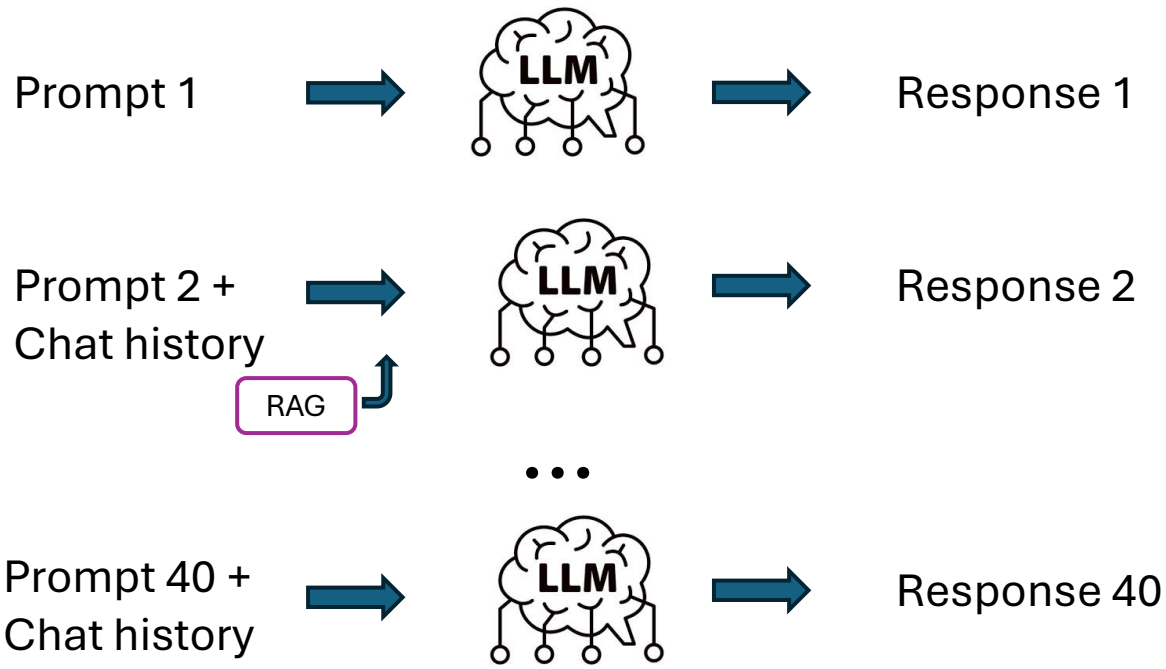
Your very long conversation with the AI

You: prompt 1
Agent: response 1
You: prompt 2
Agent: response 2
You: prompt 3
Agent: response 3
You: prompt 4
Agent: response 4
...
...
...
...
You: prompt n
Agent: response n

New friend

Best friend

Sluggish and forgetful



AI App's strategy when the context gets too big

1. Sliding window, delete old messages;
2. Compress chat history by summarizing;

Using RAG to enrich the context

What is RAG:

- RAG = **R**etrieval-**A**ugmented **G**eneration
- The agent retrieves external information and adds it to context before answering.
  Prevents guessing → improves accuracy

Good sources for RAG

- Published papers (relevant studies, methods)
- Manuals / documentation (DESeq2, STAR, etc.)
- Trusted databases (Ensembl, NCBI)
- Your own data (metadata, QC reports, results)

How to use RAG

- URL
- PDF

- **Do it early at planning phase, when writing your guardrails;**
- **It is advisable to explicitly provide RAG source.**

Avoid context contamination

Example:
If two different protocols of RNA-seq analyses are run in the same session, they might contaminate each other

---

Make sure they run in two different sessions.

`codex` This command starts a new session.

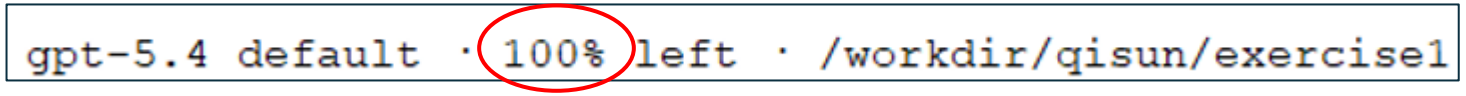
`codex resume --last` This command resume the previous session.

Be aware of token usage

- Larger context uses more tokens;
- You are charged by tokens;
- Your ChatGPT subscription plan limits tokens and context size;

You need to do your part manage the context size!

Your subscription plan has a limit on Context size

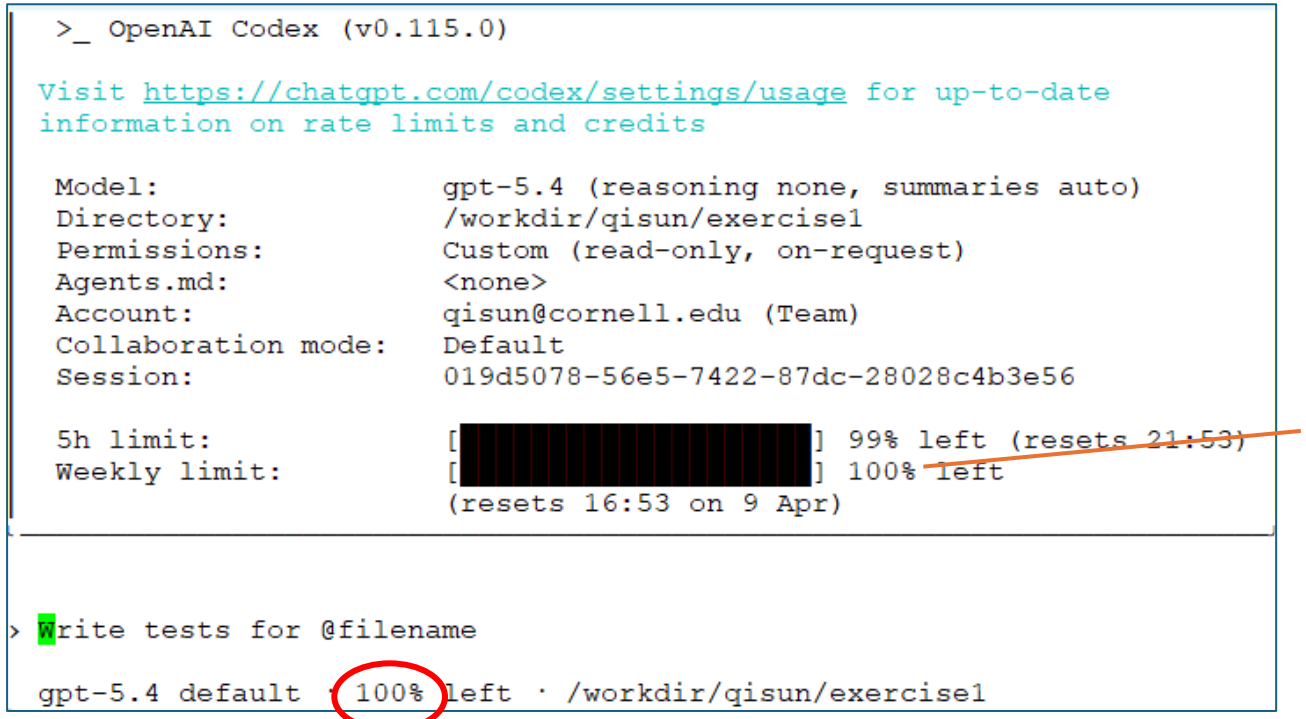


If this number is below 50%, do something.

- **Small session** -> Soft pruning/steering through prompts, e.g. ignore parts and focus on specific content.
- **Growing session** -> Run "/compact", which summarize current session.
- **Large session** -> Start a new session and provide only the relevant summary/context (you can summarize topics into files in previous session)

How to check context size and token usage on Codex?

Command: /status



- Large context, more token usage.
- If the token quota reaches 0, the agent will stop working, until reset (5-hour or 7-day cycle)

How much context memory left.

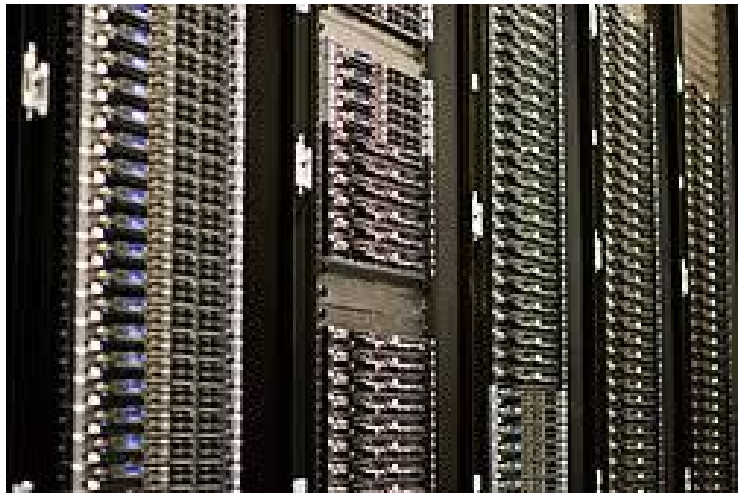
Set up and run AI agent

Check list

1. Claude or ChatGPT subscription plan;
2. Install Agent software on the server (“Codex CLI” or “Claude Code”)

AI API + LLM runs in the OpenAI or Anthropic clouds.

The agent and data analysis runs on BioHPC servers



Terminal client on your laptop

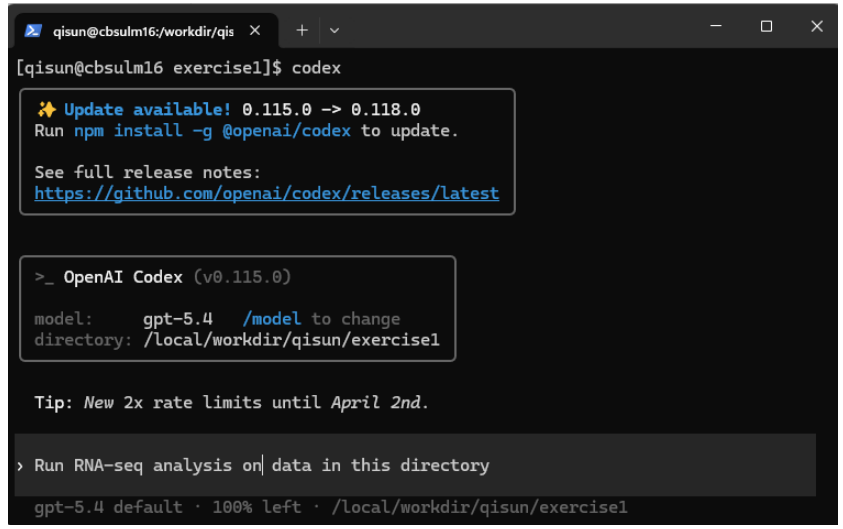
Terminals on your laptop

Mac/Windows Terminal, or any SSH client

- Ssh to the server;
- Run command: codex

(Create a project directory, and run this command from the project directory)

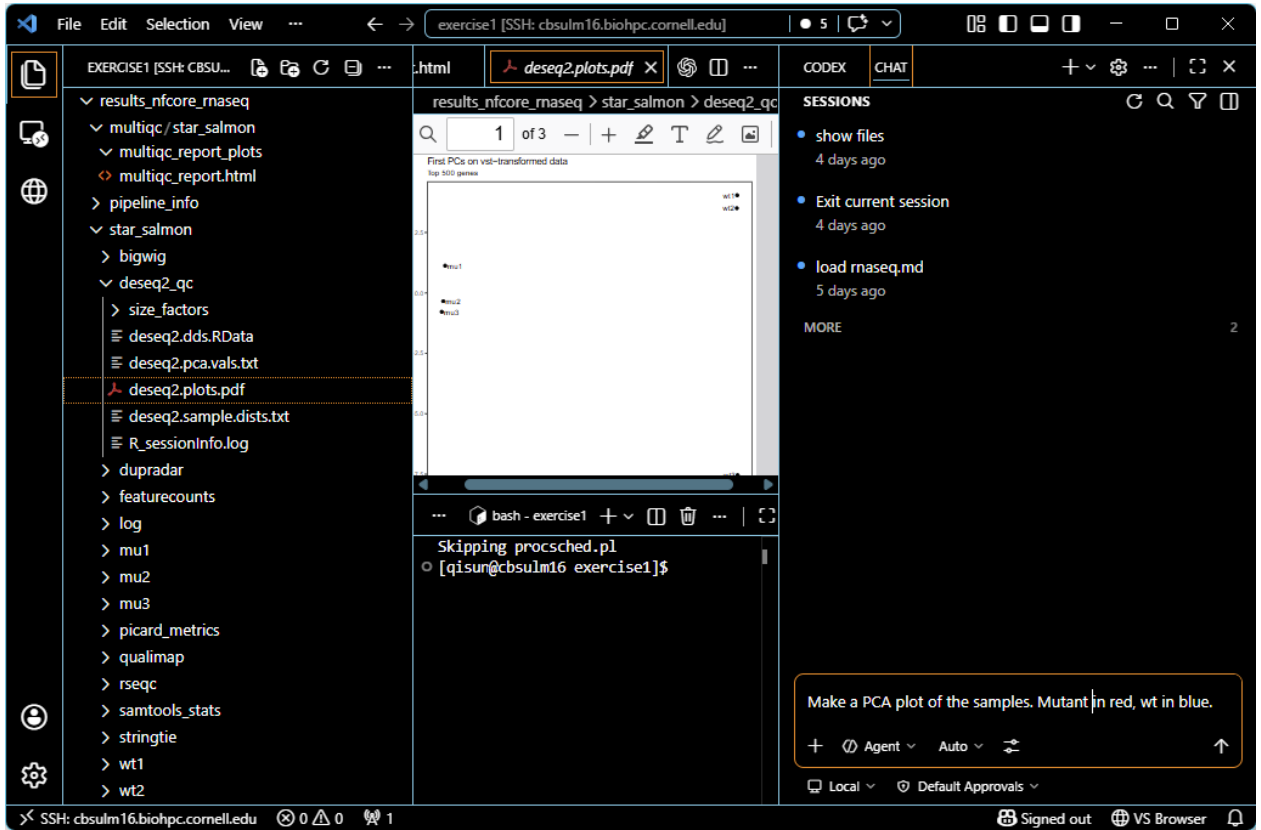
SSH terminal:



VS Code

- Install Codex extension for VS Code on your laptop;
- Open Codex inside the SSH remote session;

VS Code:



What is a Project? What is a session?

Project:

The directory where you start Codex. You can work on multiple projects, each in a separate directory.

Session:

A conversation tied to a project. You can have multiple sessions running, each in a different project.

```
codex resume --last
```

This command resume the previous session from a project. Make sure you run this command in the right directory.

What AI Agent is good at?

Read the software manuals, help you with the command syntax, and configuration file formats.

Troubleshoot if something goes wrong. “Help me to fix it”

Generate visualization plots.

Interpret the results.

Research and exploration stage

Currently, **AI-**integrated **Jupyter** or **Rstudio** cannot use the ChatGPT subscription plans. AI not available in free edition of Rstudio.

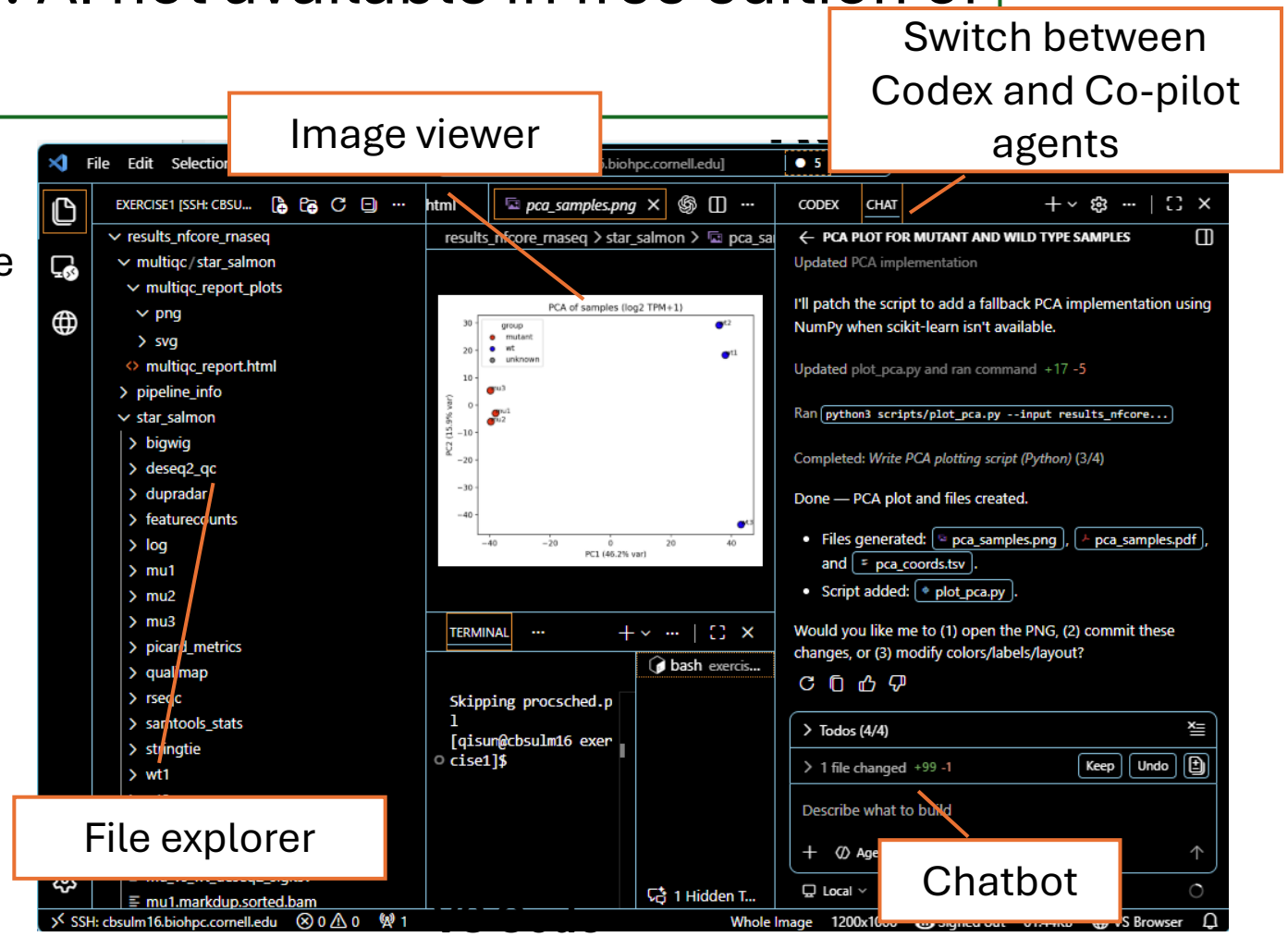
Workaround

- **Run Codex through VS Code,** which has three panels: chatbot, file explorer, image viewer.

  E.g. You can request the agent to make a PCA plot with mutants in red and wildtypes in blue. The result (PDF or png) can be viewed directly in the image panel (or right-click to download). You can request the agent to summarize the results into a report file.

Multiple agents (Codex, Co-pilot, et al) can be installed on VS Code. You can switch between the agents.

- Use Codex or ChatGPT to write code. Copy-paste the code into Jupyter or Rstudio.



AI Service Providers

Provider	Chatbot	Agent	API
OpenAI	ChatGPT	Codex CLI	Not enabled in subscription plan
Anthropic	Claude	Claude Code	
Google	Gemini	Gemini Code	
Microsoft	Co-pilot	Co-pilot	
…	…	…	…

- Pay through **subscription plans**, or **pay-by-token**.
- **Subscription plans** are heavily subsidized, **pay-by-token** is closer to the real cost.
- When quota is reached, chat may be throttled, but the agent will stop working.

Comparing Codex and Claude

- **Claude generated a better script** (better reasoning)
- **Codex is more token-efficient** (6 times more in this case).

Here are the details:

Task and setting:
- Step 1 of the hands-on project: creating a script to run RNAseq analysis.
- Same guardrail instruction file. $20/month plans for both Codex and Claude.

Token efficiency:
- **Codex** (gpt 5-4 model): use **3% of the context size limit** (18k /258K), **1% of the 5-hour token quota.**
- **Claude**(Sonnet 4.6 model): use **12% of the context size limit** (24k /200K), **6% of the 5-hour token quota.**

Script evaluation:
- Claude generated the right script. Codex got one prompt wrong. I will need to use more explicit instructions with Codex.
- Codex missed this prompt in my test: "**For QC, only do FastQC, MultiQC and Alignment stats. Skip bigWig".**
- In Claude script, it correctly skipped the other QC steps ("--skip_bigwig --skip_rseqc --skip_qualimap --skip_dupradar --skip_deseq2_qc") .
- In Codex script, it only had ("--skip_bigwig") and did not skip the other QC steps.

Better prompt correct the problem:
- Change the problematic prompt to “"**For QC, only do FastQC, MultiQC and Alignment stats, and skip the rest of the QC steps. Skip bigWig".** Both Codex and Claude produce the same correct script.

How about the free models and agent tools?

Open weight models

US
- Meta – Llama (Llama 2 / 3)

France
- Mistral – Mistral / Mixtral

China
- Alibaba – Qwen
- DeepSeek

Open-source agent framework

- Openclaw

Open-source bioinformatics agent

- Clawbio
- AutoBA

*** And a lot others**

Pros

- Model and software is free;
- More Customizable;
- Privacy

Cons

- Significant effort required to turn into functional agents.
- Expensive to maintain a GPU server.
- Privacy

Our current vision for BioHPC:

Thin AI agent layer with a robust on-premise storage and analysis infrastructure. The cost of the commercial agents should be manageable.

⚠️ AI Agent Safety

⚠️ AI Agent Safety for Data Analysis

Key risk:

AI agents (e.g., Codex, CLI agents) can **read, modify, or delete any data they can access**

🚨 What can go wrong

- Accidental file deletion (`rm -rf`, overwrites)
- Silent data corruption
- Access to unintended files (credentials, other projects)

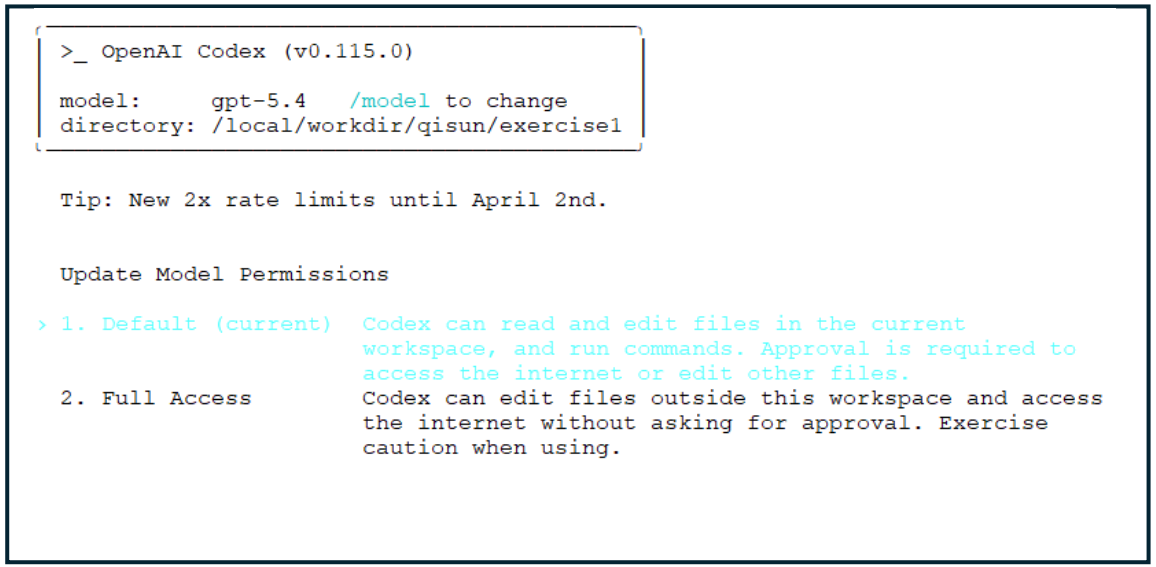
🧱 To assure file safety, you can do it the following ways (not very practical)

- Run Codex in Docker;
- Run Codex as another user

- **Check safety rules in Codex**

Run command “/permissions” in Codex. Make sure “1. Default” is selected (in blue).

Codex can read and edit files in the current workspace, and run commands. Approval is required to access the internet or edit other files.



How can Bioinformatics Facility help you?

- Maintain a **AGENTS.md** file for BioHPC system, you can copy this file to ~/.codex/.
- Working with other BRC facilities (with domain specific knowledges) to maintain guardrail files for common pipelines.
- For support, email support@biohpc.cornell.edu, or book an office hour at https://biohpc.cornell.edu/lab/office1.aspx

Acknowledgement

Cornell Bioinformatics Facility

Cornell Genomics Facility

Faraz Ahmed

Feedback

Other workshop topics?

- AI-Assisted Workflow Engineering using Nextflow.
- Other omics data analysis: Proteomics, Epigenomics, Images, Metabolomics ...