LiveDocs



You've never seen a notebook do this.

@ehsaan_iqbal

LiveDocs



You've never seen a notebook do this.

@ehsaan_iqbal

LiveDocs



Hi there! 👋

Ehsaan Iqbal

CTO, Livedocs.com

@ehsaan_iqbal

2000 Knuth: Literate Programming

2008 Jupyter: Code Meets Storytelling

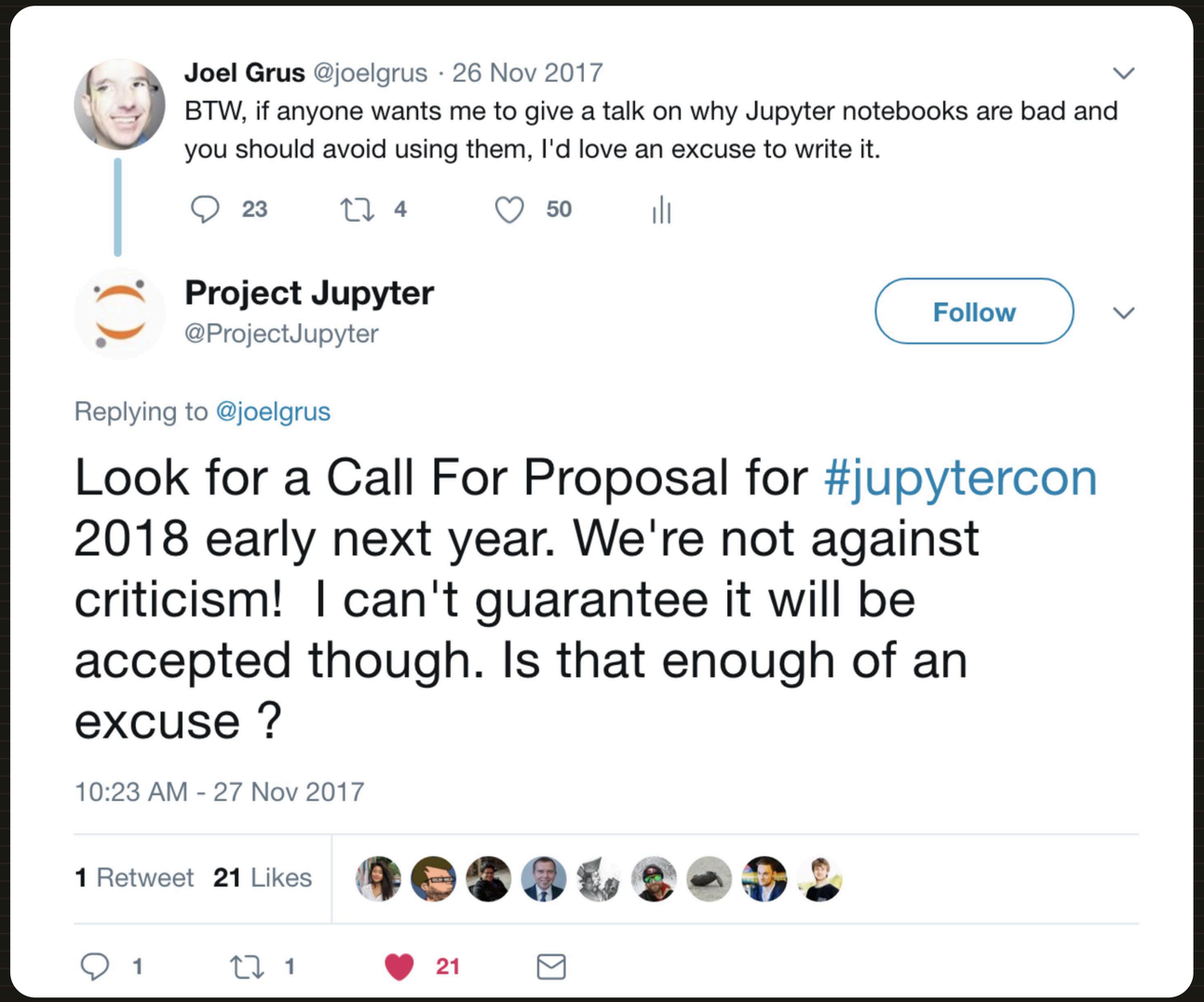
2025 Livedocs: Code Comes Alive

2000 Knuth: Literate Programming

2008 Jupyter: Code Meets Storytelling

The dark ages

2025 Livedocs: Code Comes Alive



Joel Grus @joelgrus · 26 Nov 2017

BTW, if anyone wants me to give a talk on why Jupyter notebooks are bad and you should avoid using them, I'd love an excuse to write it.

23 4 50

Project Jupyter
@ProjectJupyter

Follow

Replying to @joelgrus

Look for a Call For Proposal for #jupytercon 2018 early next year. We're not against criticism! I can't guarantee it will be accepted though. Is that enough of an excuse ?

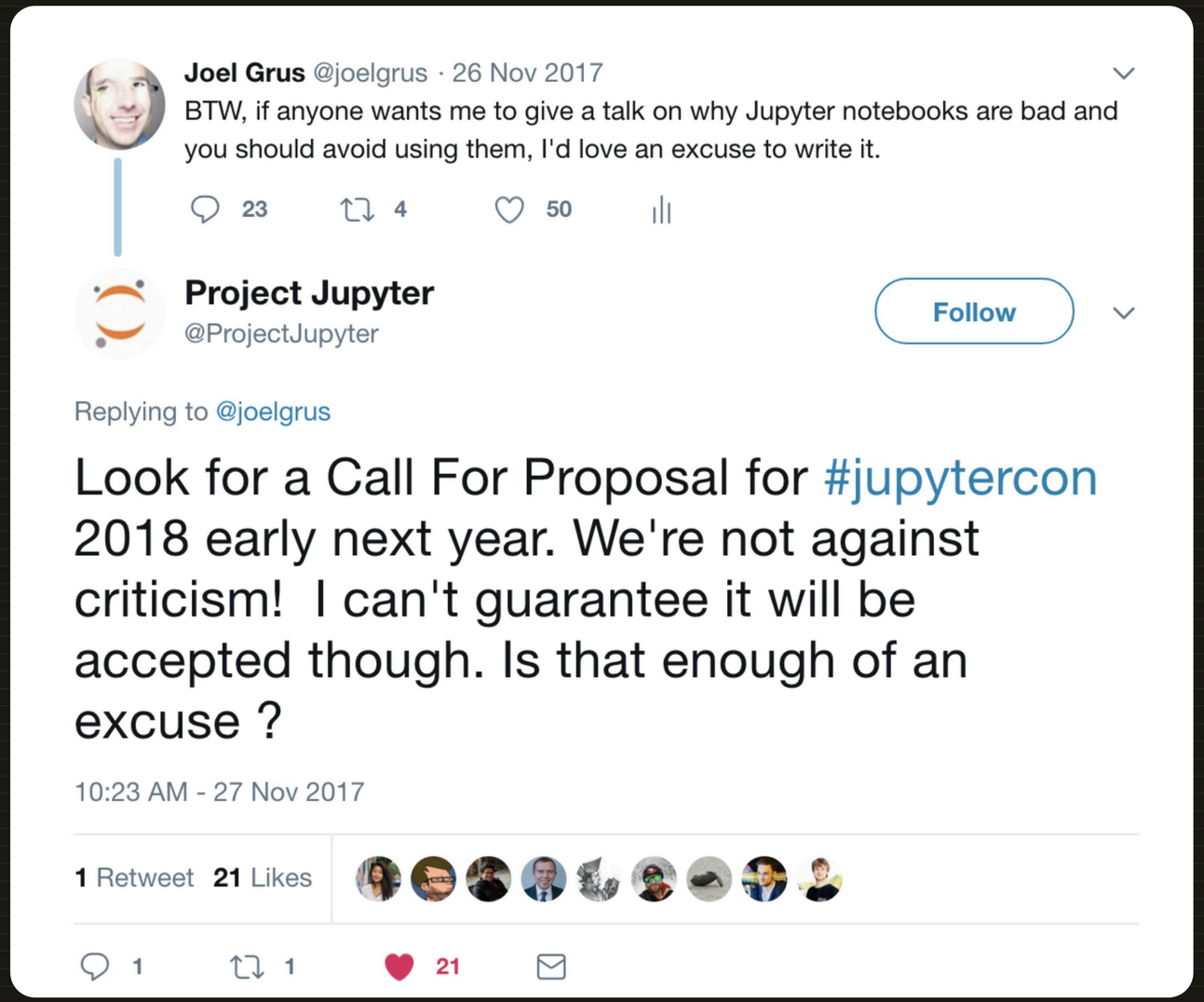
10:23 AM - 27 Nov 2017

1 Retweet **21** Likes

1 1 21

LiveDocs

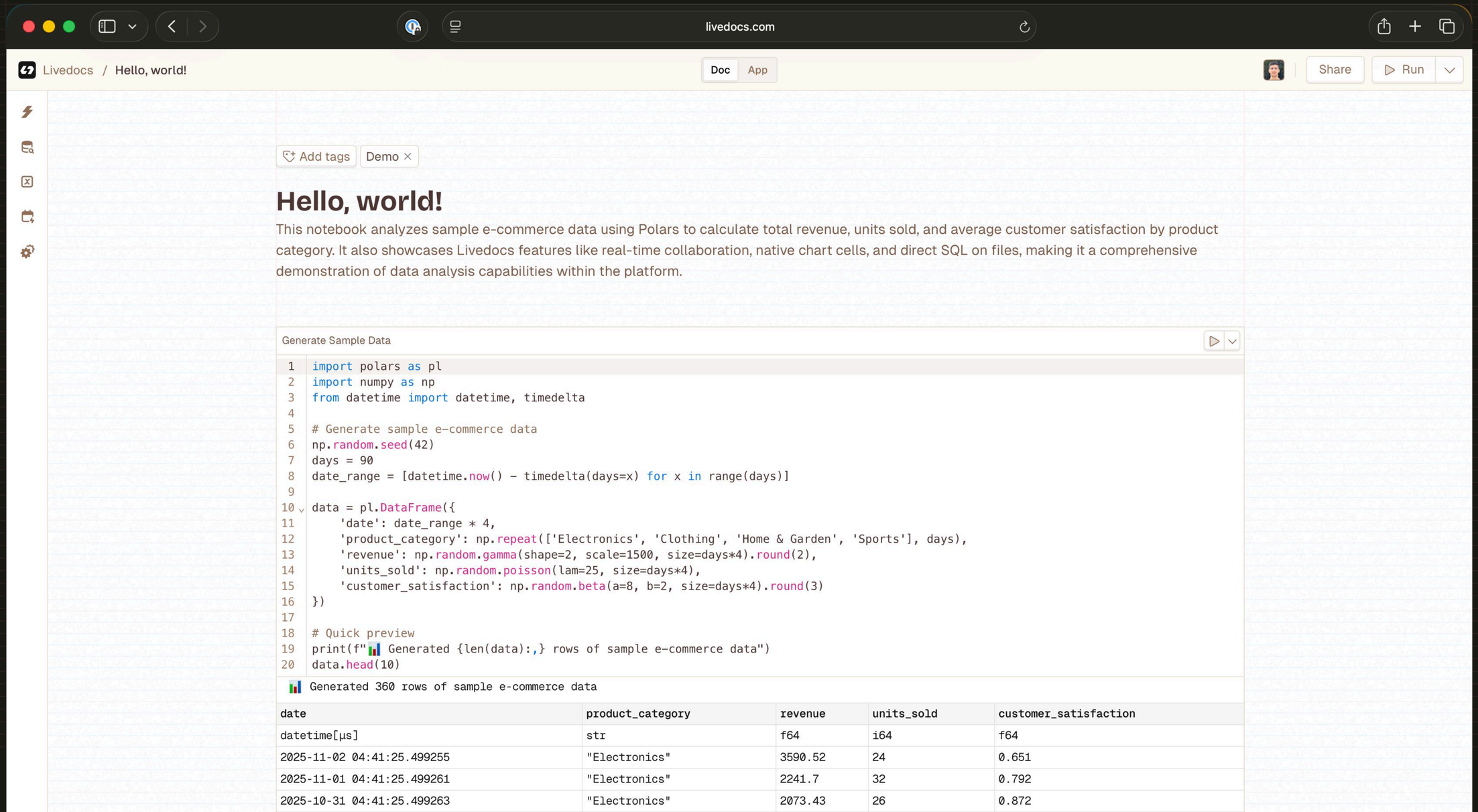
works on my machine
env drift
circular imports
mystery CSVs
missing packages
version mismatch
“final_final2.ipynb”
broken environments
dependency soup
hidden state
stale data
kernel crashes
pip vs conda chaos
outdated charts
conflicting libraries



non-deterministic results
out-of-order execution
lost variables
inconsistent outputs
no provenance
unclear lineage
no version control
merge conflicts
bloated diffs
lack of review workflow
ugly defaults
unreadable colors
limited interactivity
slow rendering
siloed analysis

and so much more!

LiveDocs


livedocs.com
Livedocs / Hello, world!
Doc
App
Share
Run
Add tags
Demo
Hello, world!
This notebook analyzes sample e-commerce data using Polars to calculate total revenue, units sold, and average customer satisfaction by product category. It also showcases Livedocs features like real-time collaboration, native chart cells, and direct SQL on files, making it a comprehensive demonstration of data analysis capabilities within the platform.
Generate Sample Data
1 import polars as pl
2 import numpy as np
3 from datetime import datetime, timedelta
4
5 # Generate sample e-commerce data
6 np.random.seed(42)
7 days = 90
8 date_range = [datetime.now() - timedelta(days=x) for x in range(days)]
9
10 data = pl.DataFrame({
11 'date': date_range * 4,
12 'product_category': np.repeat(['Electronics', 'Clothing', 'Home & Garden', 'Sports'], days),
13 'revenue': np.random.gamma(shape=2, scale=1500, size=days*4).round(2),
14 'units_sold': np.random.poisson(lam=25, size=days*4),
15 'customer_satisfaction': np.random.beta(a=8, b=2, size=days*4).round(3)
16 })
17
18 # Quick preview
19 print(f" Generated {len(data):,} rows of sample e-commerce data")
20 data.head(10)
Generated 360 rows of sample e-commerce data
date
product_category
revenue
units_sold
customer_satisfaction
datetime[μs]
str
f64
i64
f64
2025-11-02 04:41:25.499255
"Electronics"
3590.52
24
0.651
2025-11-01 04:41:25.499261
"Electronics"
2241.7
32
0.792
2025-10-31 04:41:25.499263
"Electronics"
2073.43
26
0.872

Inputs

Transform data into decisions with advanced NLP, embeddings, and machine learning—zero coding, instant outcomes. Accelerate your workflow with cutting-edge NLP and machine learning—no code, just results.

Company Name

Company Name

Acme Inc

input_1 1

User ID

User ID

5121

input_2 1

Reliability

Minimum reliability

0 1 2 3

input_3 1

Results

```
1  SELECT * FROM audit_logs.csv WHERE company_name == '{{input_1}}';
```

timestamp	user_id	action	company_name	ip_addr
2025-01-01T17:47:23.826393	51746385	EXPORT_DATA	Acme Inc	61.139.82.146
2025-01-20T08:02:30.622162	95214362	UPDATE_SETTINGS	Acme Inc	62.26.134.247
2025-01-02T05:01:12.746899	71667167	DELETE_USER	Acme Inc	95.126.206.226
2025-02-03T19:18:28.504380	95214362	UPDATE_SETTINGS	Acme Inc	236.178.229.134
2025-01-15T19:42:02.124048	71667167	VIEW_REPORT	Acme Inc	226.6.172.110
2025-01-17T21:33:16.388253	673971743	LOGIN	Acme Inc	25.40.31.161
2025-02-12T07:55:37.977398	96459761	DELETED_USER	Acme Inc	113.2.183.254

dataframe_0 1 ‹ 1 2 … 145 ›

DAG

Transform data into decisions with advanced NLP, embeddings, and machine learning—zero coding, instant outcomes. Accelerate your workflow with cutting-edge NLP and machine learning—no code, just results.

Region

Employe

Average

No more boilerplate code

Save hours every week by using Deepwork AI to write queries, build charts, and fix bugs.

Gotham ⌘+G

Sales Growth

```
SELECT
  department, COUNT(employee_id) AS employee_count,
  AVG(salary) AS avg_salary
FROM employees
GROUP BY department
GROUP BY avg_salary DESC;
```

Compare last year sales with and with out users

Instantly familiar

Schedule notebooks to run hourly, daily, weekly, or monthly.

Datasets for Growth

```
# Unique users per day
unique_users = df.groupby(df.index.date)['us
# Bounce rate
bounce_rate = df[df['pa
```

Result Collapsed

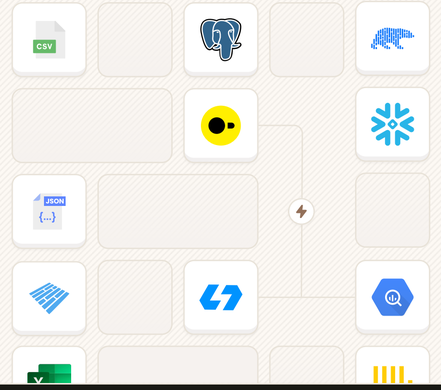
Employee data

HR

```
SELECT
 department,
 COUNT(employee_id) AS empoyee_count,
 AVG(salary) AS avg_salary
FROM
 employees
GROUP BY
 department
GROUP BY
avg_salary DESC;
```

Built for speed

Run advanced NLP, embeddings, and ML—no code, instant results.



Explain your

Run advanced NLP, e
code, instant results.

Calculate Derived Metrics

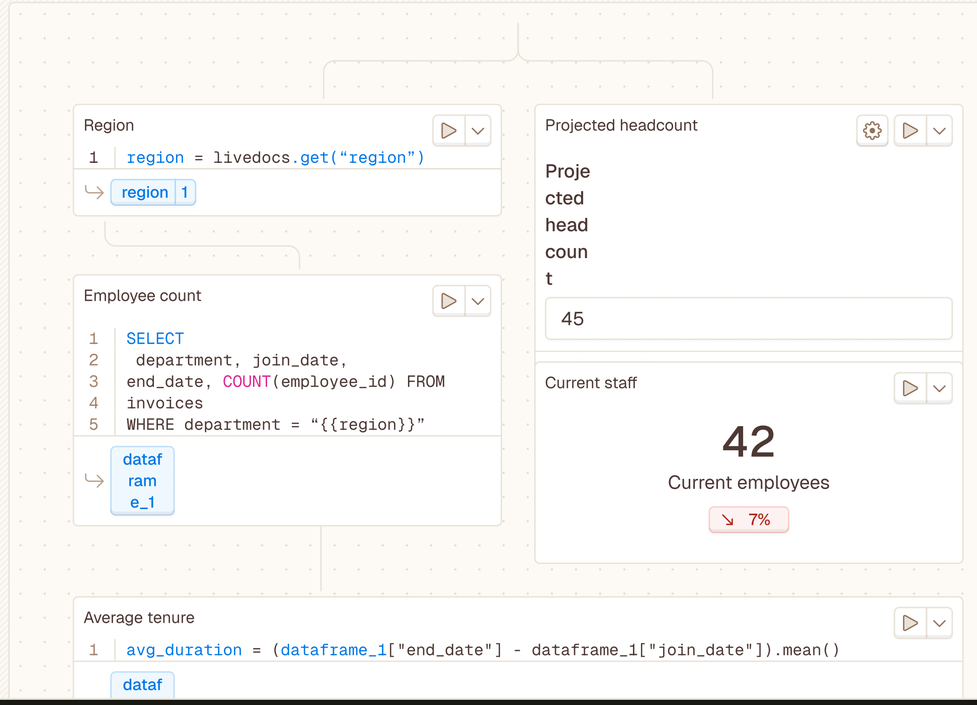
```
# Avoid divi
df["CTR"] =
df["eCPC"] =

# Check value
df[["CTR", "Conv
df = df.copy()
```

df 12

DAG

Transform data into decisions with advanced NLP, embeddings, and machine learning—zero coding, instant outcomes. Accelerate your workflow with cutting-edge NLP and machine learning—no code, just results.

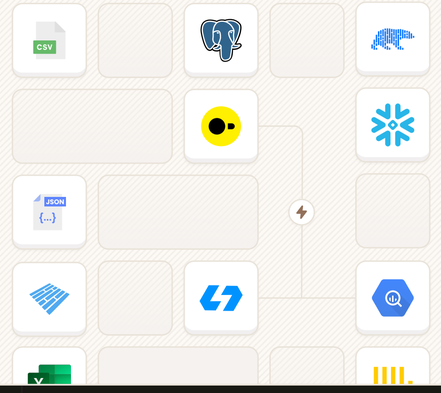


Deploy data apps

Transform data into decisions with advanced NLP, embeddings, and machine learning—zero coding, instant outcomes. Accelerate your workflow with cutting-edge NLP and machine learning—no code, just results.

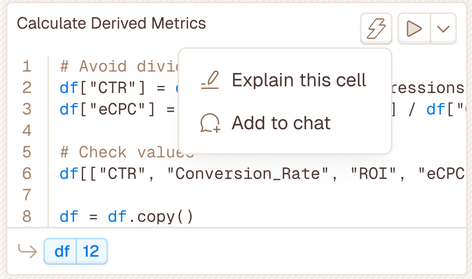
Built for speed

Run advanced NLP, embeddings, and ML—no code, instant results.



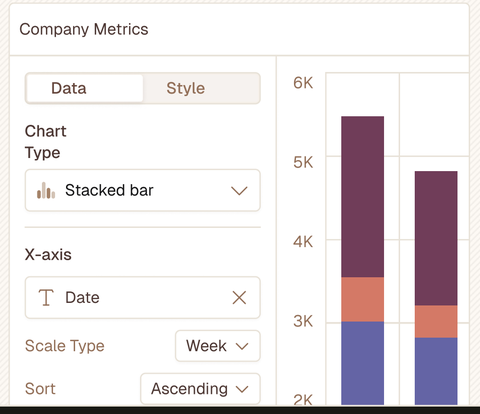
Explain your work

Run advanced NLP, embeddings, and ML—no code, instant results.



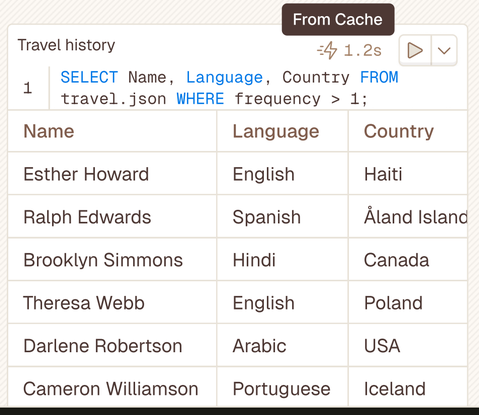
Beautiful charts

Run advanced NLP, embeddings, and ML—no code, instant results.



Cache

Run advanced NLP, embeddings, and ML—no code, instant results.



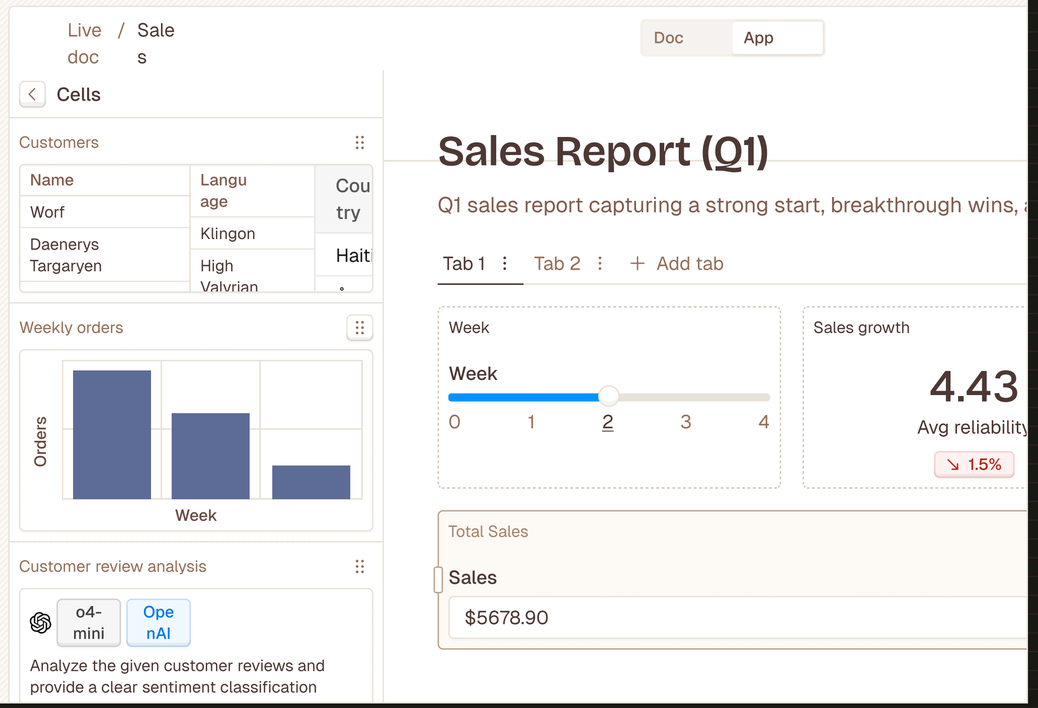
Secret

Run advan
code, inst

@ehsaan_iqbal

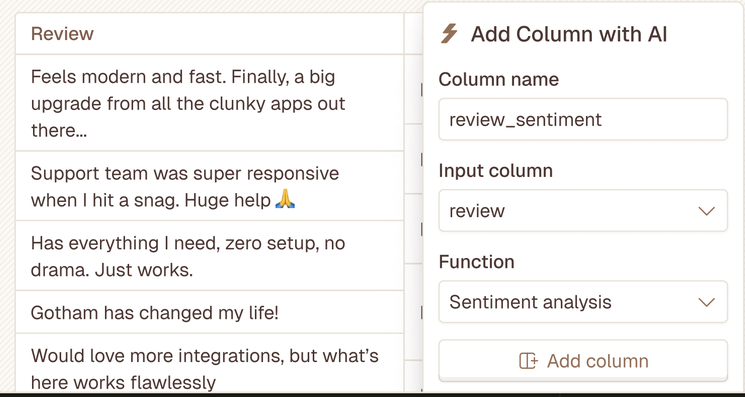
Deploy data apps

Transform data into decisions with advanced NLP, embeddings, and machine learning—zero coding, instant outcomes. Accelerate your workflow with cutting-edge NLP and machine learning—no code, just results.



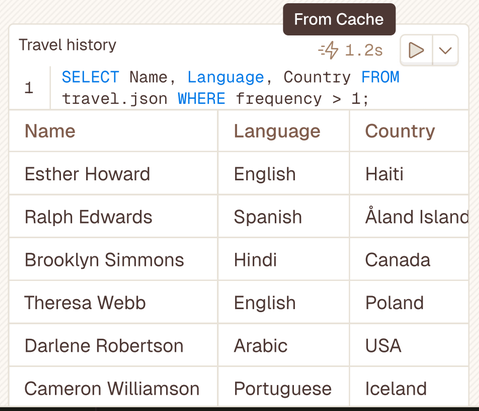
Effortless AI

Run advanced NLP, embeddings, and ML—no code, instant results.



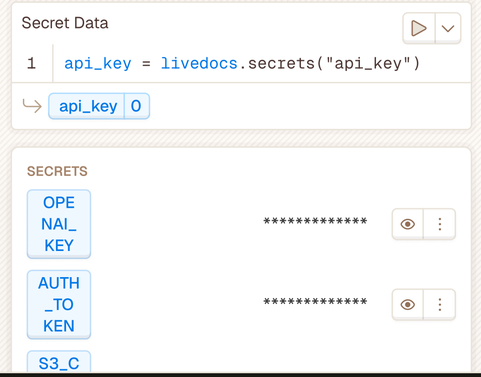
Cache

Run advanced NLP, embeddings, and ML—no code, instant results.



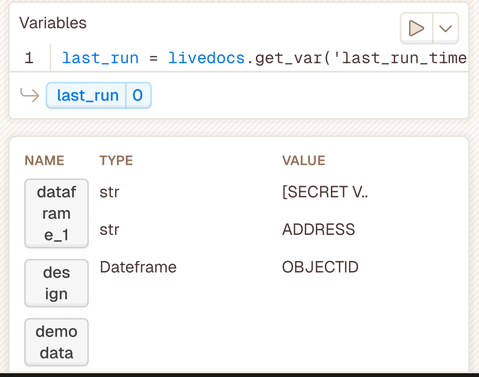
Secrets

Run advanced NLP, embeddings, and ML—no code, instant results.



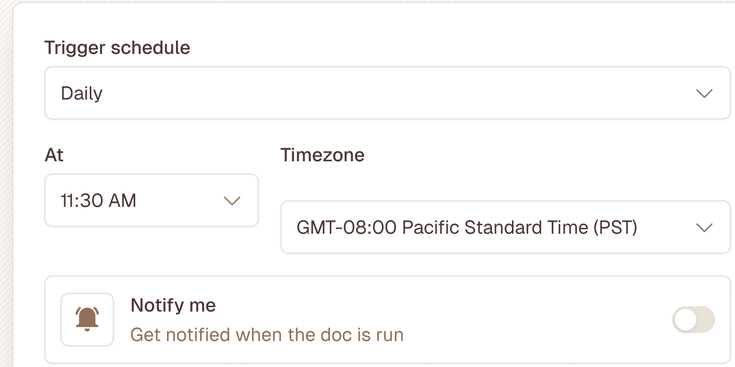
Variables

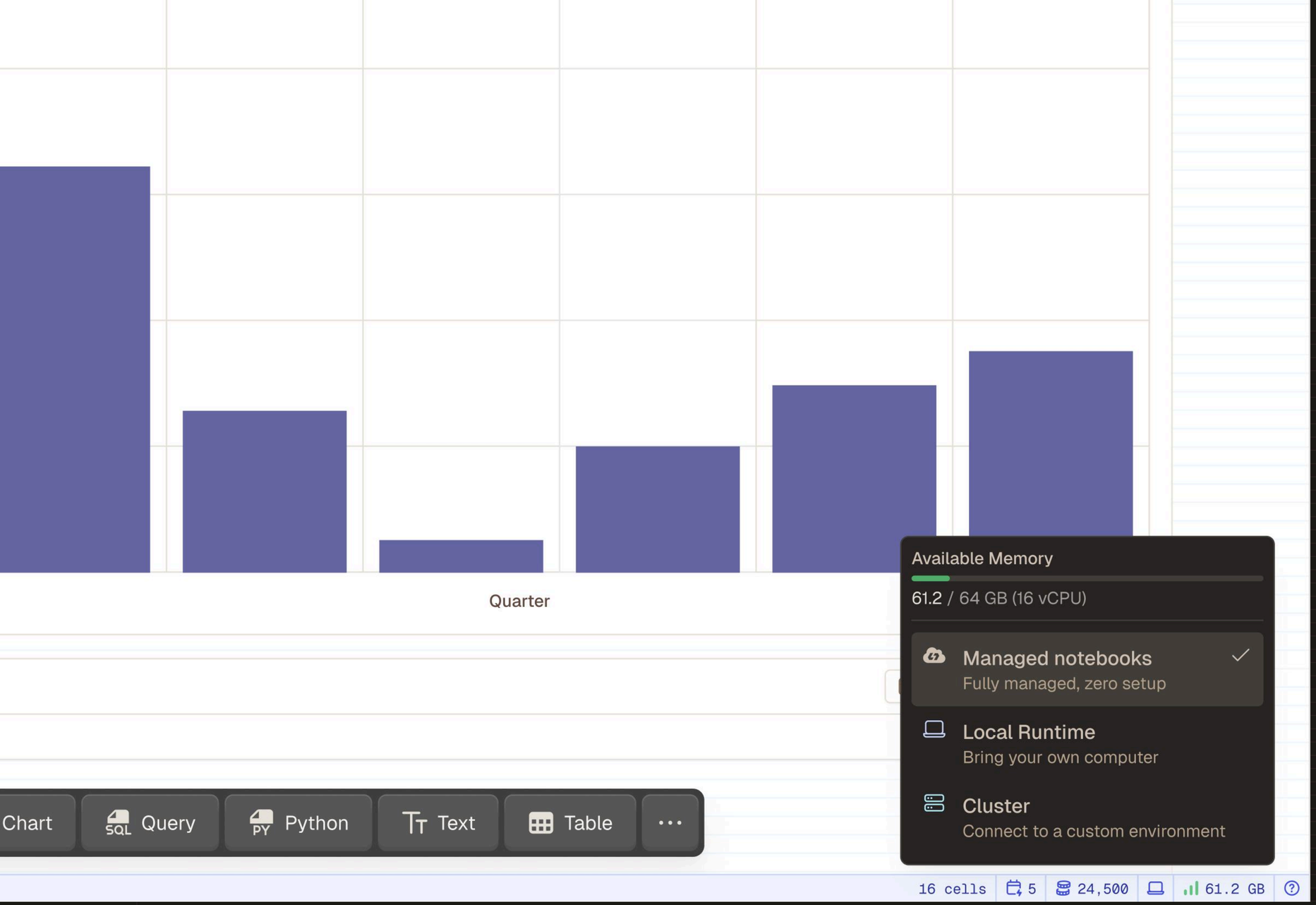
Run advanced NLP, embeddings, and ML—no code, instant results.



Schedule notebooks

Schedule notebooks to run hourly, daily, weekly, or monthly.



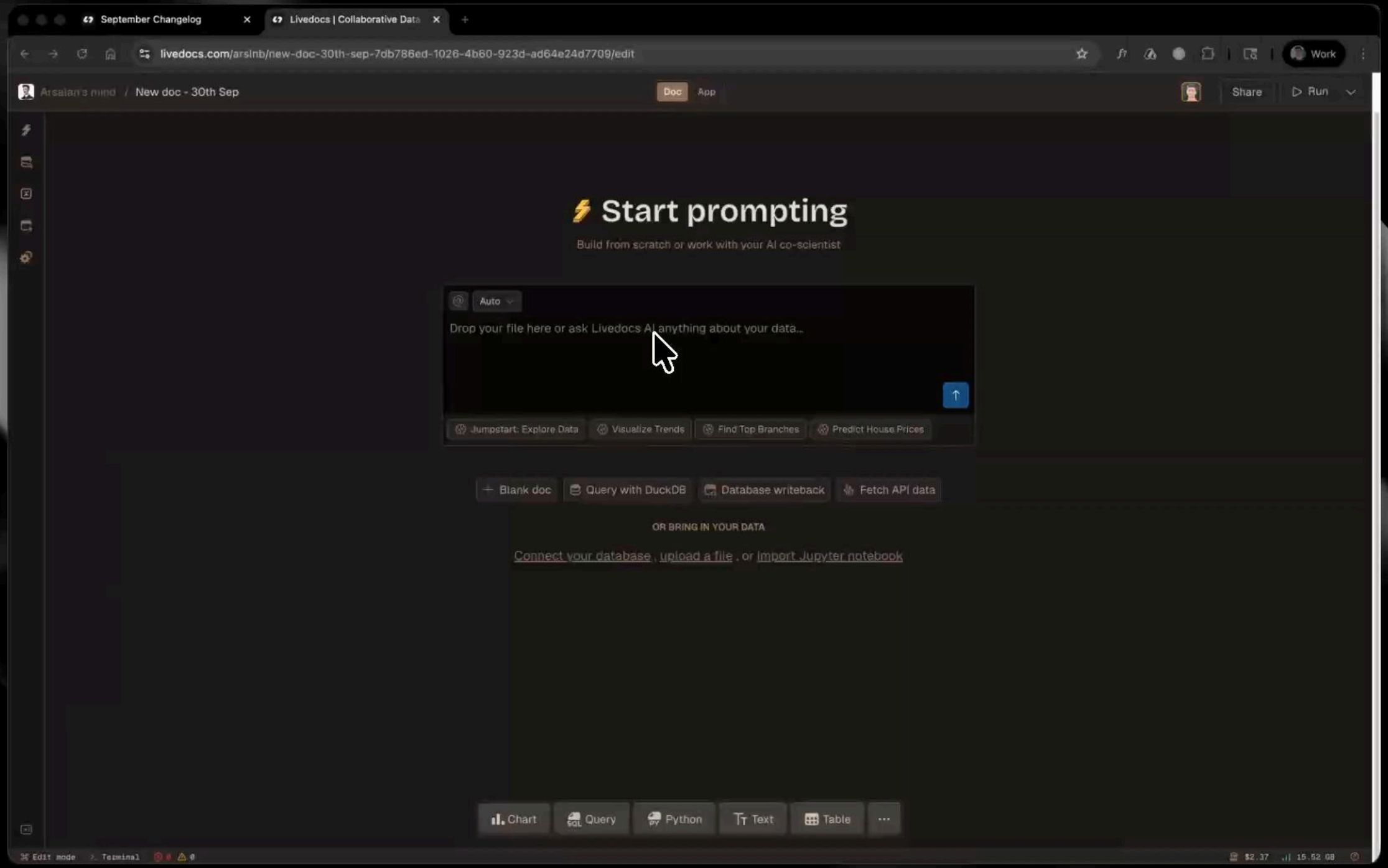


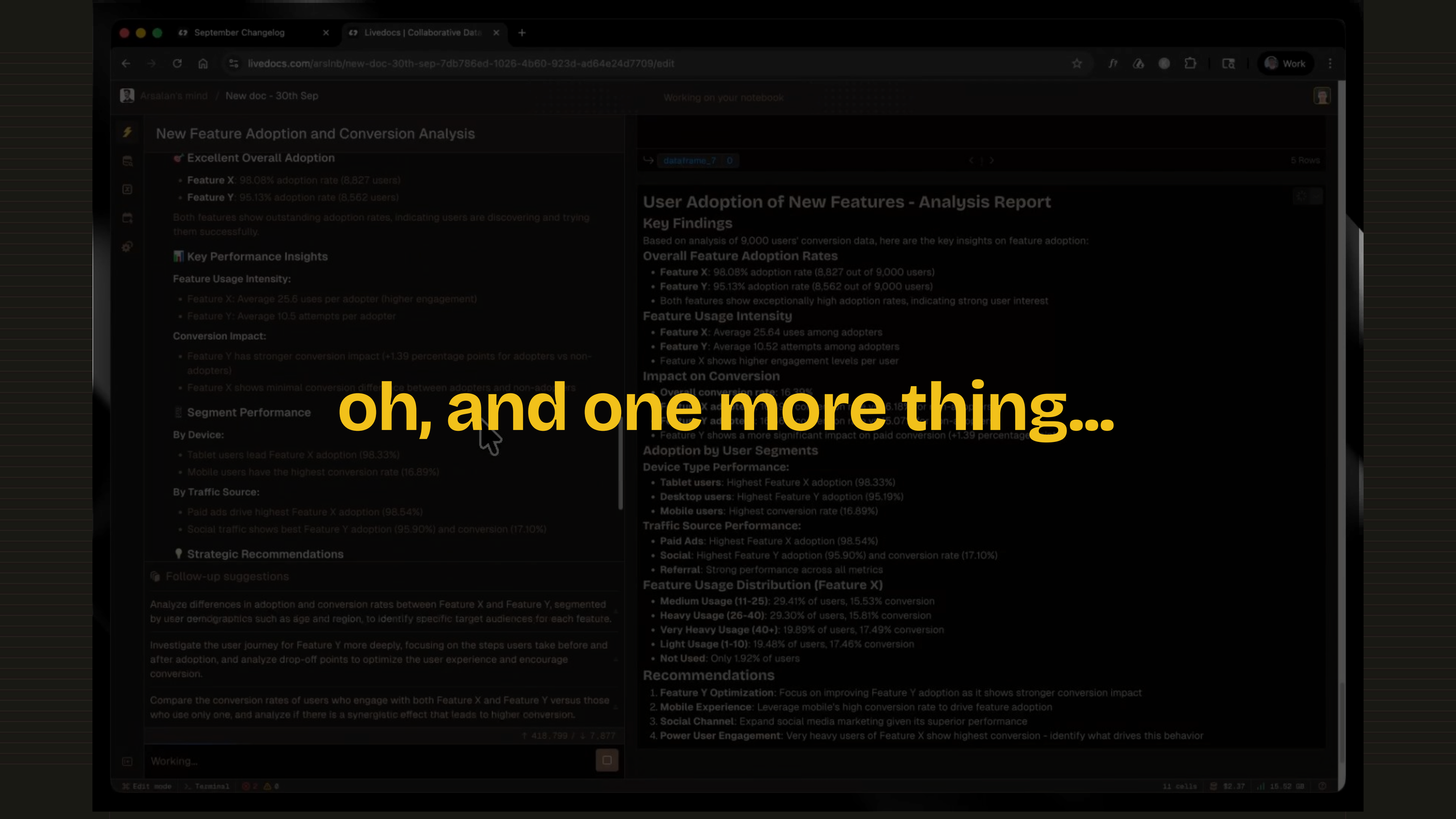
best of both worlds

Livedocs anywhere = Livedocs everywhere

@ehsaan_iqbal Bash

```
$ livedocs launch
```


September Changelog
Livedocs | Collaborative Data
livedocs.com/arslnb/new-doc-30th-sep-7db786ed-1026-4b60-923d-ad64e24d7709/edit
Work
Arsalan's mind / New doc - 30th Sep
Doc
App
Share
Run
Start prompting
Build from scratch or work with your AI co-scientist
Auto
Drop your file here or ask Livedocs AI anything about your data...
Jumpstart: Explore Data
Visualize Trends
Find Top Branches
Predict House Prices
Blank doc
Query with DuckDB
Database writeback
Fetch API data
OR BRING IN YOUR DATA
Connect your database , upload a file , or import Jupyter notebook
Chart
Query
Python
Text
Table
Edit mode
Terminal
$2.37
15.52 GB


September Changelog
Livedocs | Collaborative Data
livedocs.com/arslnb/new-doc-30th-sep-7db786ed-1026-4b60-923d-ad64e24d7709/edit
Work
Arsalan's mind / New doc - 30th Sep
Working on your notebook
New Feature Adoption and Conversion Analysis
Excellent Overall Adoption
Feature X: 98.08% adoption rate (8,827 users)
Feature Y: 95.13% adoption rate (8,562 users)
Both features show outstanding adoption rates, indicating users are discovering and trying them successfully.
Key Performance Insights
Feature Usage Intensity:
Feature X: Average 25.6 uses per adopter (higher engagement)
Feature Y: Average 10.5 attempts per adopter
Conversion Impact:
Feature Y has stronger conversion impact (+1.39 percentage points for adopters vs non-adopters)
Feature X shows minimal conversion difference between adopters and non-adopters
Segment Performance
By Device:
Tablet users lead Feature X adoption (98.33%)
Mobile users have the highest conversion rate (16.89%)
By Traffic Source:
Paid ads drive highest Feature X adoption (98.54%)
Social traffic shows best Feature Y adoption (95.90%) and conversion (17.10%)
Strategic Recommendations
Follow-up suggestions
Analyze differences in adoption and conversion rates between Feature X and Feature Y, segmented by user demographics such as age and region, to identify specific target audiences for each feature.
Investigate the user journey for Feature Y more deeply, focusing on the steps users take before and after adoption, and analyze drop-off points to optimize the user experience and encourage conversion.
Compare the conversion rates of users who engage with both Feature X and Feature Y versus those who use only one, and analyze if there is a synergistic effect that leads to higher conversion.
↑ 418,799 / ↓ 7,877
Working...
dataframe_7 0
5 Rows
User Adoption of New Features - Analysis Report
Key Findings
Based on analysis of 9,000 users' conversion data, here are the key insights on feature adoption:
Overall Feature Adoption Rates
Feature X: 98.08% adoption rate (8,827 out of 9,000 users)
Feature Y: 95.13% adoption rate (8,562 out of 9,000 users)
Both features show exceptionally high adoption rates, indicating strong user interest
Feature Usage Intensity
Feature X: Average 25.64 uses among adopters
Feature Y: Average 10.52 attempts among adopters
Feature X shows higher engagement levels per user
Impact on Conversion
Feature Y shows a more significant impact on paid conversion (+1.39 percentage
Adoption by User Segments
Device Type Performance:
Tablet users: Highest Feature X adoption (98.33%)
Desktop users: Highest Feature Y adoption (95.19%)
Mobile users: Highest conversion rate (16.89%)
Traffic Source Performance:
Paid Ads: Highest Feature X adoption (98.54%)
Social: Highest Feature Y adoption (95.90%) and conversion rate (17.10%)
Referral: Strong performance across all metrics
Feature Usage Distribution (Feature X)
Medium Usage (11-25): 29.41% of users, 15.53% conversion
Heavy Usage (26-40): 29.30% of users, 15.81% conversion
Very Heavy Usage (40+): 19.89% of users, 17.49% conversion
Light Usage (1-10): 19.48% of users, 17.46% conversion
Not Used: Only 1.92% of users
Recommendations
1. Feature Y Optimization: Focus on improving Feature Y adoption as it shows stronger conversion impact
2. Mobile Experience: Leverage mobile's high conversion rate to drive feature adoption
3. Social Channel: Expand social media marketing given its superior performance
4. Power User Engagement: Very heavy users of Feature X show highest conversion - identify what drives this behavior
oh, and one more thing...

LiveDocs

LiveDocs

Lets jam!

