

Jupyter notebook's pdf export feature is not as rich as necessary in some cases, particularly with regards to maths exports etc. however it is common to want to include notebook source and output into a LaTeX document.

In that case working from LaTeX and importing individual notebook cells is most effective and provides the nicest resulting document. To that end `ipynb-tex` provides new TeX commands to include cells directly from notebooks.

1 Command Examples

1.1 Markdown Content

Simple markdown content can be included directly in the document:

```
\IpynbSource{notebook.ipynb}{intro}
```

This text came from notebook.ipynb

1.2 Source Code

The source code cell can be exported directly in :

```
\begin{ minted }[
    breaklines,
    escapeinside=||
]{python}
| \IpynbSource{notebook.ipynb}{tag1} |
\end{ minted }
```

```
for i in range(3):
    print(i)
```

Source can include TeXcode

```
\IpynbSource{notebook.ipynb}{equation}
```

Simpson's rule is a method of numeric integration, as opposed to analytics methods of integration, specifically it is this approximation:

$$\int_a^b f(x) \, dx \approx \frac{b-a}{6} (f(a) + 4f\left(\frac{a+b}{2}\right) + f(b))$$

Cell Output

Include the output of a cell

```
\IpynbOutText{notebook.ipynb}{tag1}
```

0

1

2

Include the output of a cell with \TeX in it

```
\IpynbOutLatex{notebook.ipynb}{tag3}
```

$$(1, \quad 2)$$

[In progress] Handle sympy

```
\IpynbSource{notebook.ipynb}{sympy}
$$\ipynboutput{notebook.ipynb}{sympy}$$
```

1.3 Images

Include an image, technically ipynb-texonly outputs a base-64 encoded image, but it can be included, by using Christian Sachs' `luainageembed` package.

```
\includegraphicseembedded[width=0.5\pagewidth]{
  \IpynbOutImage{notebook.ipynb}{graph}
}
```

