PYCON ODESSA

mindsdb

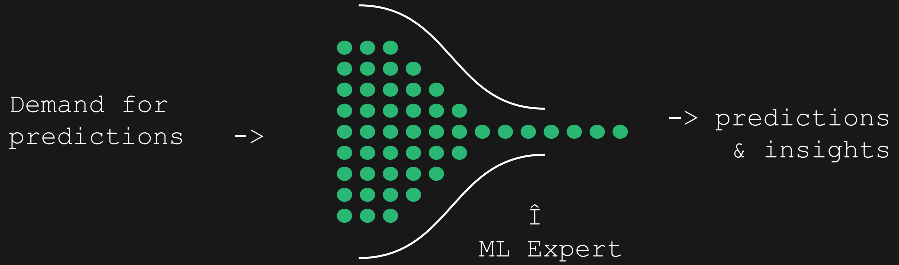
Democratizing Machine Learning

mindsdb

Why M.L?

- Use the data that we have to make informed predictions.

Why Democratizing M.L?



Note: True ML experts are rare and on high demand [1].

[1] https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=IML14576USEN&

How to Democratize M.L?

- Simple ways for other domain experts (non ML people) to exploit ML on their own.

Demand for predictions -> -> predictions & insights

First step?

- Empower Developers that are NOT ML Experts.

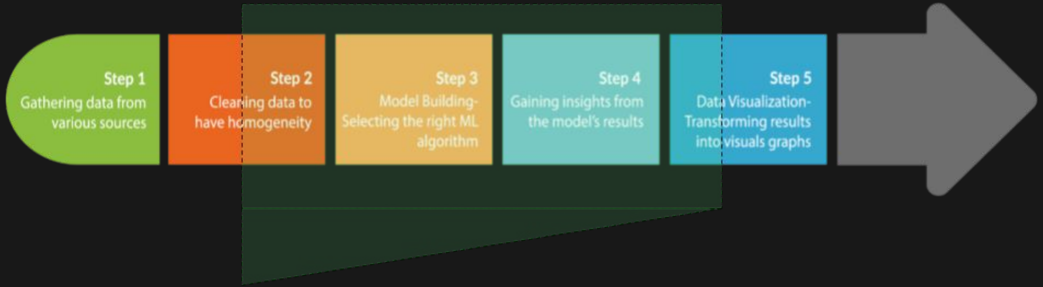
Why?

- Developers are AWESOME! we build shit, we get it done!
- Developers have easier access to data and means to aggregate it.
- There are many more developers than ML experts out there.

mindsdb

How?

- Abstract complexity and expose it via tools developers already know.



Step 1
Gathering data from various sources

Step 2
Cleaning data to have homogeneity

Step 3
Model Building-Selecting the right ML algorithm

Step 4
Gaining insights from the model's results

Step 5
Data Visualization-Transforming results into visuals graphs

Abstract this complexity

What’s important

- X.A.I: Explainable AI.

Why?

- If we are delegating the ML complexity to a system, we need to trust the system.

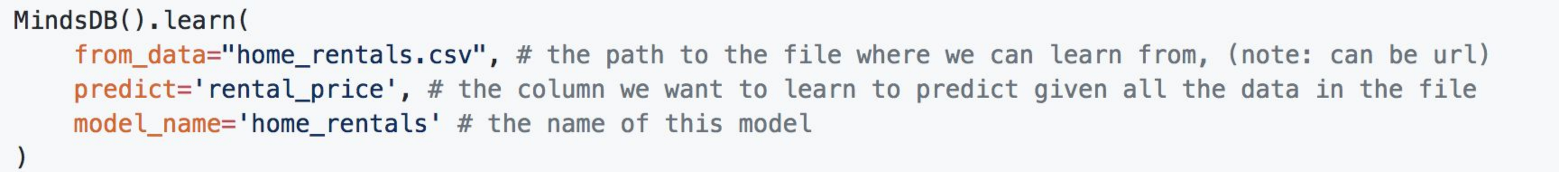
mindsdb

Our approach

WWW.MINDSDB.COM

Goal 1: Frictionless

- Be able to create and use complex predictive models with no more than a line of code.



```
MindsDB().learn(
    from_data="home_rentals.csv", # the path to the file where we can learn from, (note: can be url)
    predict='rental_price', # the column we want to learn to predict given all the data in the file
    model_name='home_rentals' # the name of this model
)
```

mindsdb

Goal 2: X.A.I Ready

Must answer:

- Why this prediction?
- Why not something else?
- When is the model working?
- When is the model not working?
- When can I trust the model?
- When can’t I trust the model?
- How can I make my models better

mindsdb

Goal 3: Stay State Of Art

- ML/AI moves fast so do we.

mindsdb

```
pip3 install mindsdb
```

Thank you!

:)

mindsdb

```
pip3 install mindsdb
```

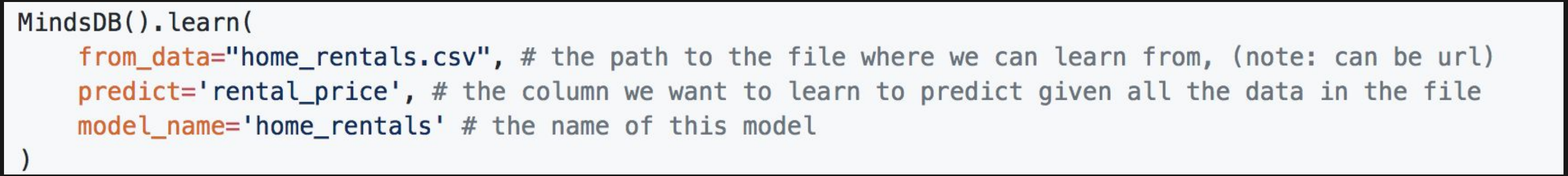
Just kidding, not done yet

What is coming this month?

MindsDB V.1.0

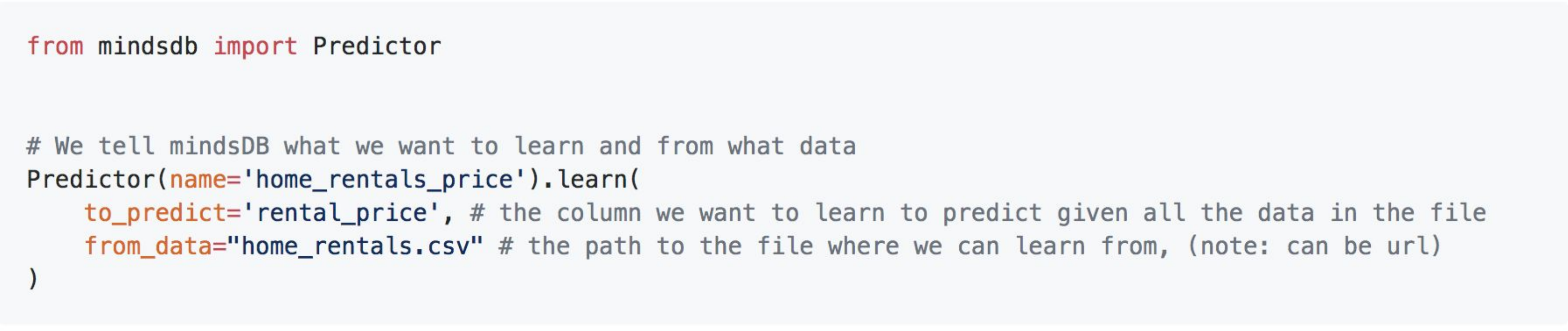
- Cleaner API
- Many more data types
- Tensorflow/Pytorch you pick
- XAI (Quality domain)
    - Data quality
    - Model quality
    - Prediction quality
- Production system

mindsdb > Cleaner API



```
MindsDB().learn(
    from_data="home_rentals.csv", # the path to the file where we can learn from, (note: can be url)
    predict='rental_price', # the column we want to learn to predict given all the data in the file
    model_name='home_rentals' # the name of this model
)
```

Introducing Predictor objects



```
from mindsdb import Predictor


# We tell mindsDB what we want to learn and from what data
Predictor(name='home_rentals_price').learn(
    to_predict='rental_price', # the column we want to learn to predict given all the data in the file
    from_data="home_rentals.csv" # the path to the file where we can learn from, (note: can be url)
)
```

PYCON ODESSA

mindsdb > More Data types

From:

- NUMERIC
- CATEGORICAL

To:

- NUMERIC
  - INT
  - FLOAT
- CATEGORICAL
  - SINGLE
  - MULTIPLE
- DATETIME
  - STRING
  - TIMESTAMP
  - TIMESTAMP_DELTA
- BINARY
  - IMAGE
  - AUDIO
  - VIDEO
- SEQUENTIAL
  - TEXT
  - NESTED 1
    - NUMERIC
    - DATETIME
    - PAIRS

      # example pairs of [**[NUMERIC, TIMESTAMP_DELTA]**, ….]

PYCON ODESSA

mindsdb > XAI

From:

- None :(

To:

```
predictor.explain_quality()

prediction.explain_quality()
```

mindsdb > **XAI**

predictor.explain_quality()

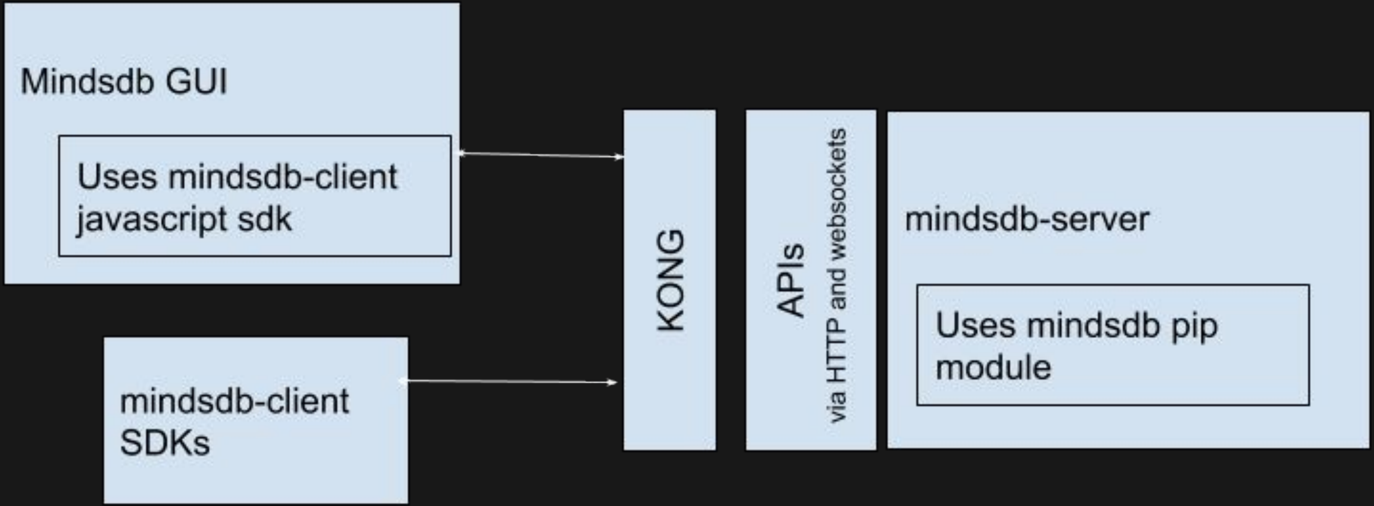
- **DATA QUALITY**
  - Evaluate quality of the data that you train from AND tell you about what we find:
    - **Consistency:** How much error
    - **Redundancy:** Repeated data in columns and rows
    - **Variability:**
      - Biases
      - Noise/outliers
      - No variation
    - **Completeness:** How much missing data
- **MODEL QUALITY**
  - What values its **good at** predicting
  - What values its **not good at** predicting
    - What data quality issues are responsible for this

mindsdb > XAI

```
prediction.explain_quality()
```

- Predicted value
- Certainty of predicted value
  - Probability distribution of possible values

mindsdb > From desktop to production



mindsdb

Thank you

This time for real, bye ;)
Ah! And ask me about early access to all this