Data Camp Live Training:
Machine Learning with XGBoost

DataCamp

Lis Sulmont

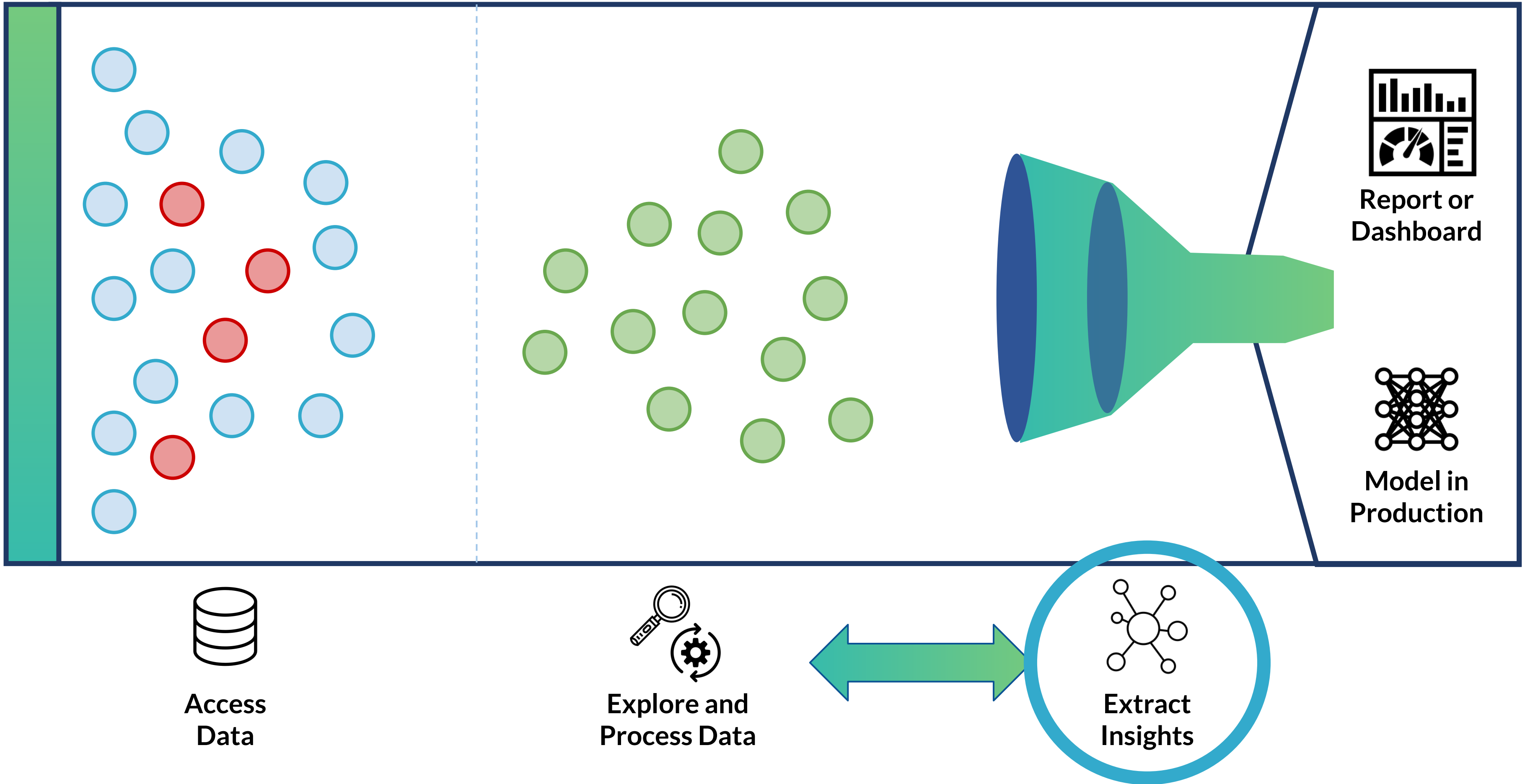
Curriculum Manager, DataCamp

DataCamp

Session Agenda

- *Intro and recap on gradient boosting*
- Getting to know our data
- Q & A
- Your First XGBoost Classifier
- Q & A
- Cross Validation in XGBoost
- Digging into Parameters
- Q & A
- Hyperparameter tuning
- Q & A
- Take home assignment
- Recap/Closing Notes

The data science workflow - where will our focus be today?



Dataset Overview

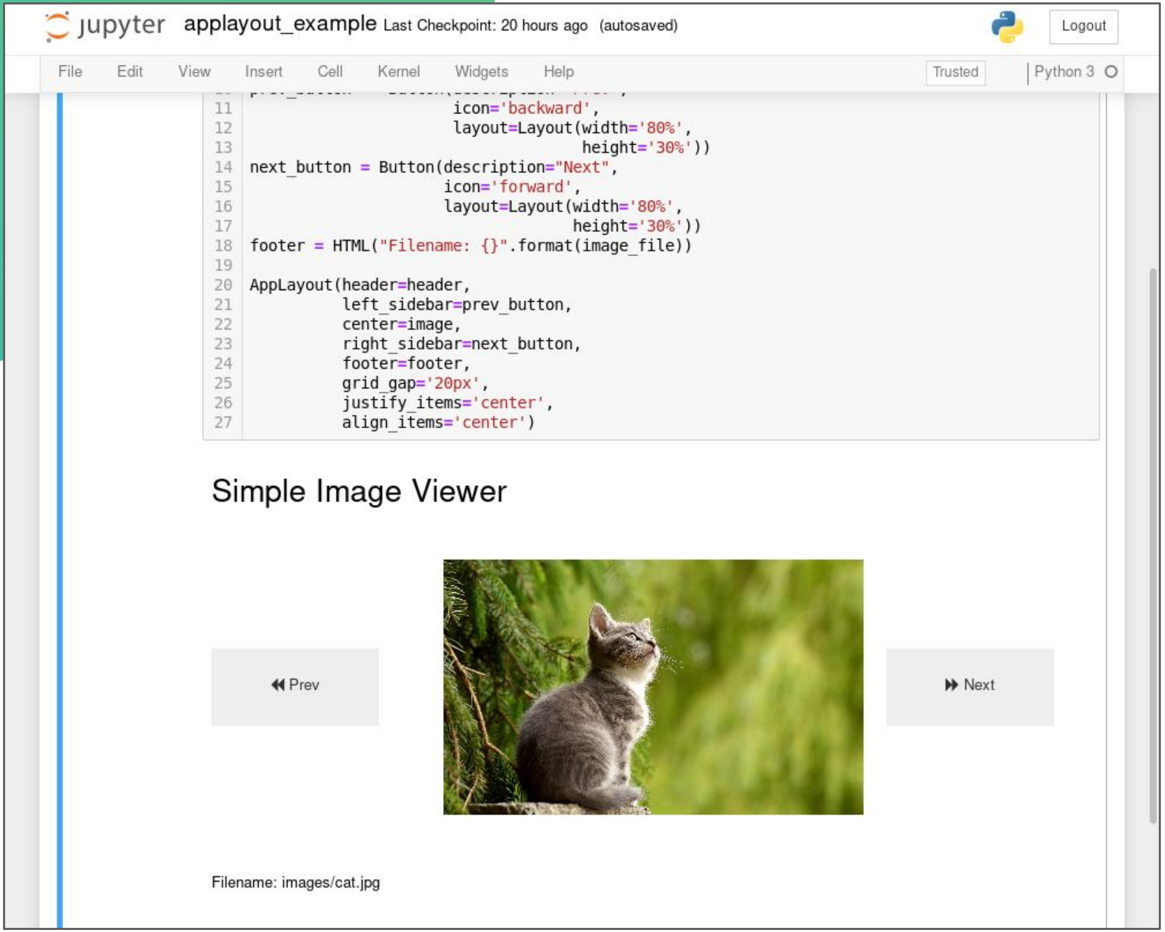
Dataset: **Hotel Booking Demands**

Problem:

Can we predict if a booking will be cancelled?

53 pre-processed columns including:

- ***is_cancelled:*** *Binary variable indicating whether a booking was canceled*
- ***lead time:*** *Number of days between booking date and arrival date*
- ***avg_daily_rate:*** *Average daily rate*
- ***deposit_type_No Deposit:*** *Binary variable indicating whether a deposit was made*
- ***booked_by_agent:*** *Binary variable indicating whether the booking was booked by an agent*
- ***stays_in_weekend_nights:*** *Number of weekend nights booked*
- ***previous_cancellations:*** *Number of prior bookings that were cancelled by the customer*



```python
prev_button = Button(description="Prev",
                      icon='backward',
                      layout=Layout(width='80%',
                                    height='30%'))
next_button = Button(description="Next",
                     icon='forward',
                     layout=Layout(width='80%',
                                   height='30%'))
footer = HTML("Filename: {}".format(image_file))

AppLayout(header=header,
          left_sidebar=prev_button,
          center=image,
          right_sidebar=next_button,
          footer=footer,
          grid_gap='20px',
          justify_items='center',
          align_items='center')
```

Simple Image Viewer

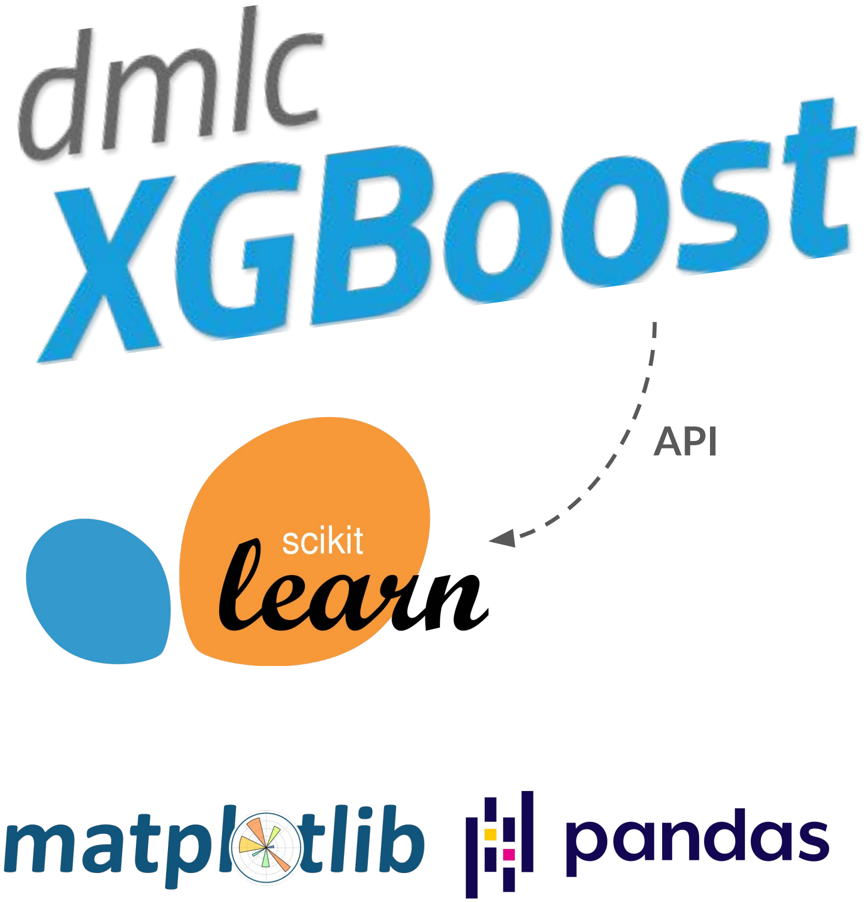
Filename: images/cat.jpg

❗ ❗ **Requires a gmail account to edit** ❗ ❗

Originally written in

XGBoost is available in several languages.

We'll be using the Python package.



XGBoost

eXtreme Gradient Boosting

“XGBoost is an optimized distributed gradient boosting library designed to be **highly efficient, flexible and portable.** It implements machine learning algorithms under the **Gradient Boosting framework.** ” [1]

Why use XGBoost?

Gained popularity from Kaggle dominance

State of the art performance on ML competitions and outperforms single-algorithm methods [2]

Scikit-learn compatible API

Also integrates well with R’s caret and data flow frameworks like Apache Spark and Hadoop.

Speed and performance

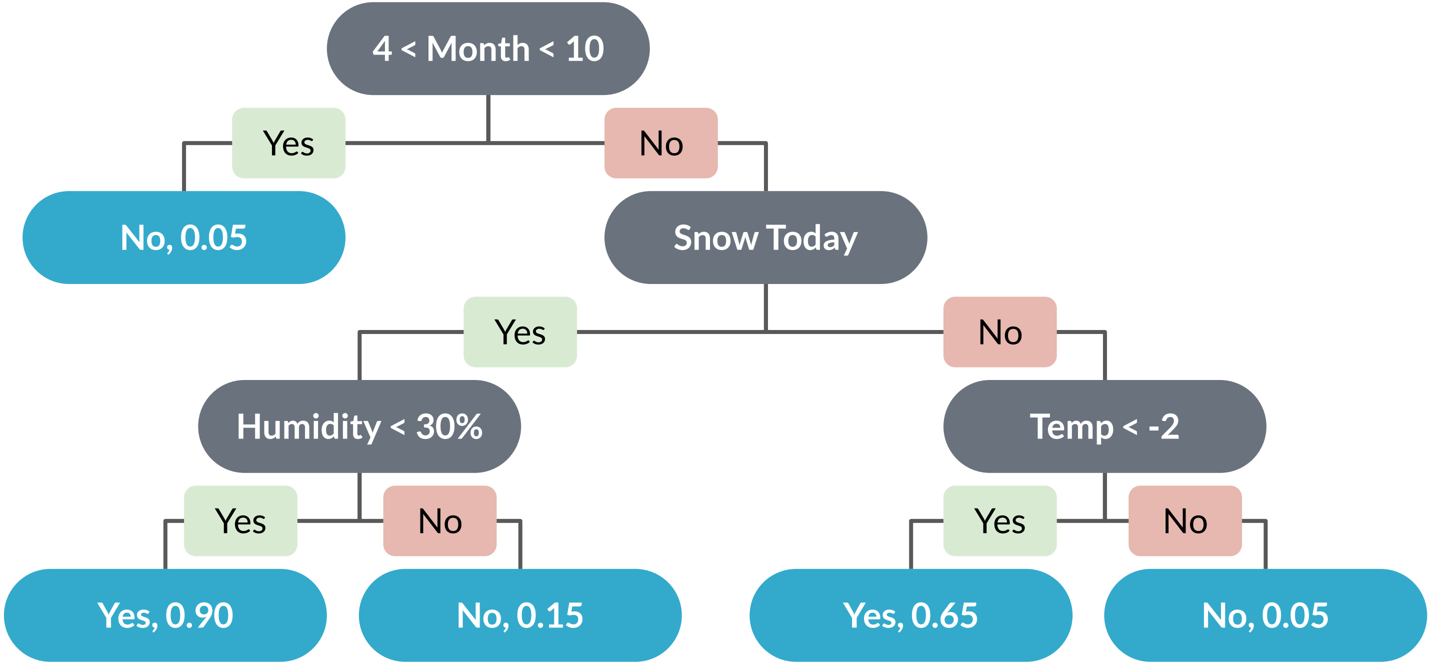
Faster and more scalable implementation of **gradient boosting**

Gradient Boosting Recap

Decision Trees

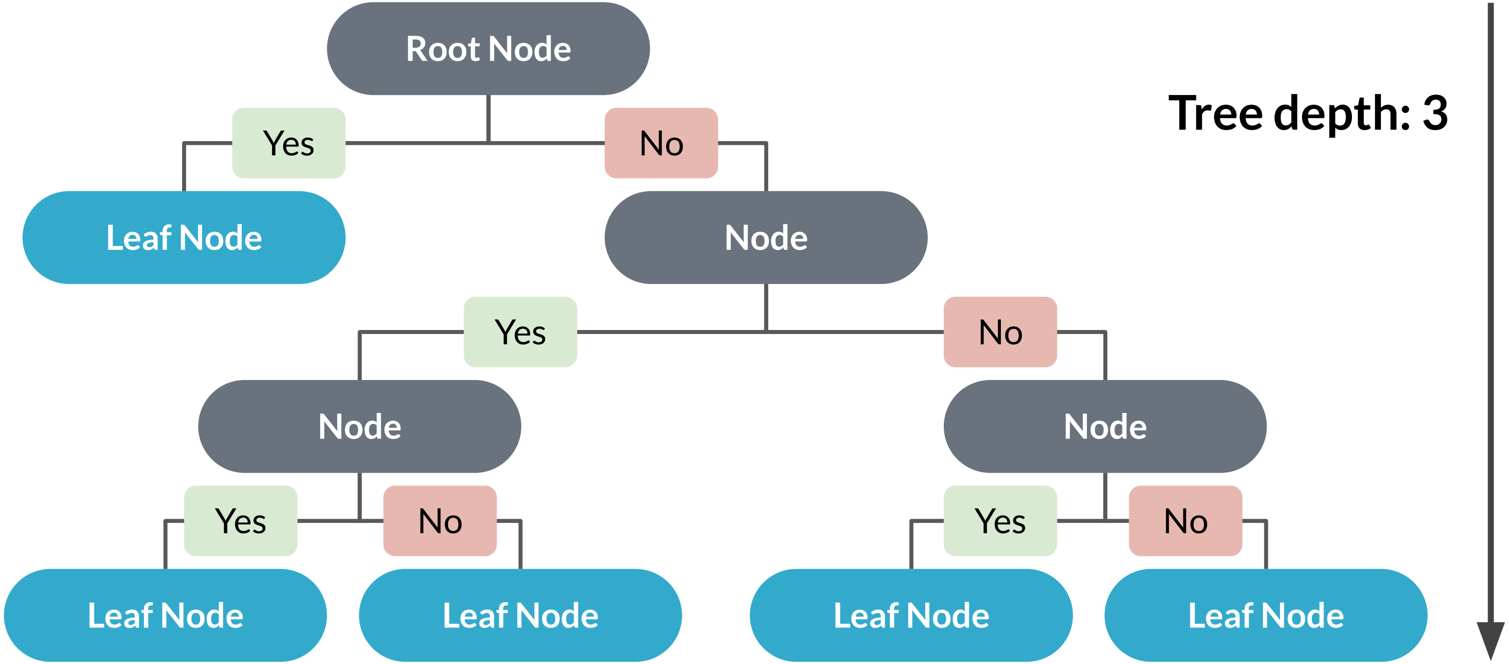
- Machine learning technique that uses tree structures
- At each decision node, the data is split into **two** based on a feature
- Split by finding the best information gain possible
- Constructed iteratively until stopping criteria is met -> leaf node
- Works for regression and classification problems
- Classification and Regression Trees **(CART)**
    - Each leaf node contains a prediction score, not only the decision

Example: Will it snow tomorrow?



Threshold: 0.5

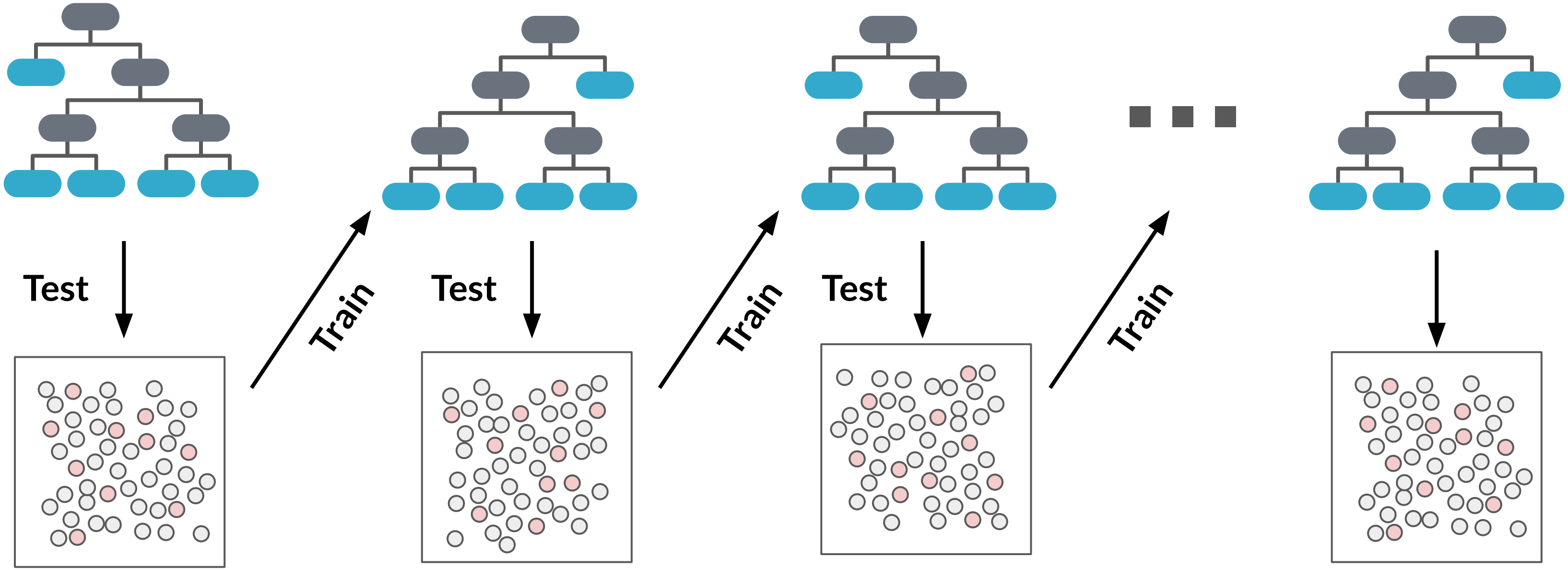
Tree Terminology



Boosting

- **Ensemble method:** convert many weak learners into a strong learner
  - Weak learners = slightly better than chance
  - Decision trees are great weak learners
- Boosting is accomplished by
  - Sequentially train weak learners to correct its predecessor
  - Each weak prediction weighed according to performance
  - Combine the weighted predictions to get a single weighted prediction

Gradient Boosting



Fit next predictor to the predecessor's residual errors

XGBoost Weak Learners

aka base learners, boosters

- Decision tree (most common)
  - `booster=gbtree`
- Generalized linear regression
  - `booster=gblinear`

Session Notebook

Recap and Closing Notes

What Did We Learn Today?

Implementing gradient boosting models with XGBoost

Tuning XGBoost parameters

Using Scikit-Learn with XGBoost

How to further improve performance

More boosting rounds

We only went up to 40 in this session!

More time for hyperparameter optimization

Grid search, random search, Bayesian optimization. Check out our Hyperparameter Tuning in Python course!

Learn about other tunable parameters

Complete our Extreme Gradient Boosting with XGBoost!

XGBoost for regression

Scikit-Learn API

- `xgboost.XGBRegressor()` class

Choose an appropriate loss function

- E.g., `objective=reg:squarederror`

When to use gradient boosting

Supervised machine learning

- # of features < # of training samples
- Mix of categorical and numerical features
- Or just numerical features

Not good at deep learning tasks

Large feature space, e.g., computer vision and natural language processing

Coming Soon!

Don’t miss these upcoming webinars and live training sessions!

- Machine Learning with Scikit Learn (6/30)
- Brand Analysis using Social Media Data in R (7/2)

Take Home Question

Take Home Question

What is the highest accuracy you can reach on the test set (`X_test`,`y_test`) after training on the training set (`X_train`,`y_test`)?

Make sure to play around with the parameters and their values in rs_param_grid.

Submission options:

- Share your code snippet and output on LinkedIn - make sure to tag DataCamp and me!
- Send me your code snippet and output by email (lis@datacamp.com)
- Tag us `@DataCamp` with the hashtag `#datacamplive`

Thank you

Lis Sulmont
Curriculum Manager, DataCamp
lis@datacamp.com
www.linkedin.com/in/elisabethsulmont/