Things to do while you are waiting

- Course slides are available at: https://hprc.tamu.edu/training/aces_intel.html
- Get ready to SSH to the FASTER cluster
  - For ACCESS users:
    - Disconnect from your non-TAMU VPN
  - For TAMU users:
    - Log into TAMU VPN (if you're off campus)

Data Science for Python

using the FASTER and ACES clusters
in preparation for Intel AI Analytics Toolkit

by Richard Lawrence

Date: 10/25/2022

Fall 2022

Outline

- Getting Started with FASTER and ACES
- Jupyter Notebook Environment
- Data Structure with Pandas
- Machine Learning with Scikit Learn
- Machine Learning with XGBoost

Learning Resources

- ACCESS Documentation https://access-ci.atlassian.net/wiki/spaces/ACCESSdocumentation/pages/95915115/FASTER+Texas+A+M
- HPRC Wiki https://hprc.tamu.edu/wiki/FASTER
- HPRC on Youtube https://www.youtube.com/c/TexasAMHPRC

Getting Started with FASTER and ACES

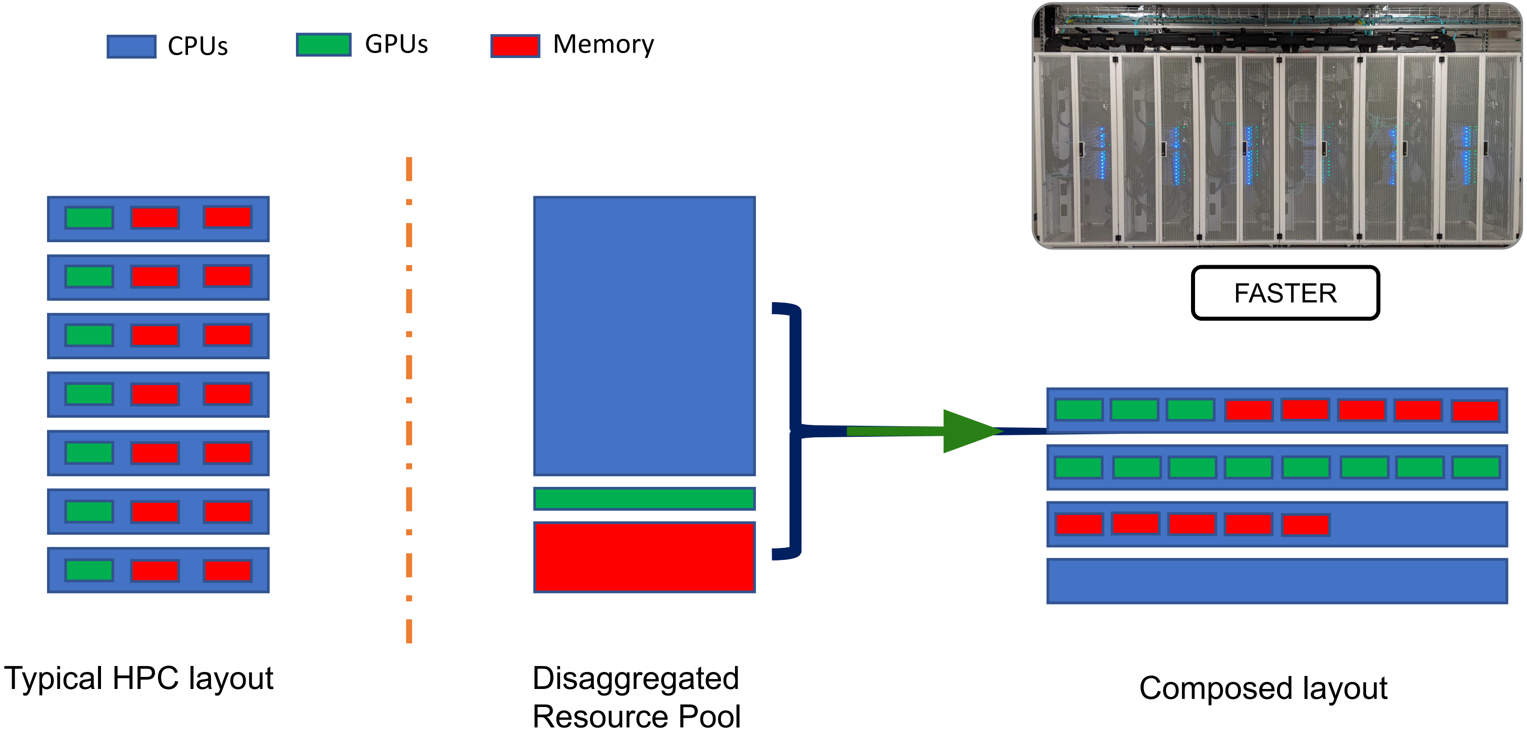
FASTER Cluster

hprc.tamu.edu/wiki/FASTER:Intro

Node Type	Quantity
64-core login nodes	4 (3 for TAMU, 1 for ACCESS)
64-core compute nodes (256GB RAM each)	180 (11,520 cores)
Composable GPUs	200 T4 16GB 40 A100 40GB 10 A10 24GB 4 A30 24GB 8 A40 48GB
Interconnect	Mellanox HDR100 InfiniBand (MPI and storage) Liqid PCIe Gen4 (GPU composability)
Global Disk	5PB DDN Lustre appliances

FASTER (Fostering Accelerated Sciences Transformation Education and Research) is a 180-node Intel cluster from Dell featuring the Intel Ice Lake processor.

Composability at the Hardware Level



hprc.tamu.edu/resources

ACES - Accelerating Computing for Emerging Sciences (Phase I)

Component	Quantity	Description
Graphcore IPU	16	16 Colossus GC200 IPUs and dual AMD Rome CPU server on a 100 GbE RoCE fabric
Intel FPGA PAC D5005	2	FPGA SOC with Intel Stratix 10 SX FPGAs, 64 bit quad-core Arm Cortex-A53 processors, and 32GB DDR4
Intel Optane SSDs	8	3 TB of Intel Optane SSDs addressable as memory using MemVerge Memory Machine.

ACES Phase I components are available through FASTER

Overview: Jupyter Lab on FASTER

1. Reach a login node
2. Make a copy of the exercise files
3. Reach a compute node
4. Open Jupyter Lab in browser

Accessing FASTER via SSH (TAMU users)

Two-Factor Authentication enabled using TAMU CAS.

- Off campus:
  - Set up and start VPN (Virtual Private Network): u.tamu.edu/VPnetwork
- SSH programs for Windows:
  - MobaXTerm (preferred, includes SSH and X11)
  - PuTTY SSH
  - Windows Subsystem for Linux

hprc.tamu.edu/wiki/HPRC:Access

Accessing FASTER for TAMU users

- FASTER has two login nodes for TAMU users.
- SSH to either login node:

```
ssh -L <useridnum>:localhost:<useridnum>
netid@faster.hprc.tamu.edu
```

Accessing FASTER for ACCESS users

- ACCESS users must submit their ssh public key for installation in the FASTER jump host.
- FASTER has 1 login node for ACCESS users.
- SSH to login node via Jump Host:

```
$ ssh
-L <useridnum>:localhost:<useridnum>
-J <fasterusername>@faster-jump.hprc.tamu.edu:8822
<fasterusername>@login.faster.hprc.tamu.edu
```

Files for the Exercises

- Navigate to your personal scratch directory
  `$ cd $SCRATCH`
- Files for this course are located at
  `/scratch/training/intel-aiml-aces`
  Make a copy in your personal scratch directory
  `$ cp -r /scratch/training/intel-aiml-aces $SCRATCH`
- Enter this directory (your local copy)
  `$ cd intel-aiml-aces`
- Make a copy of the Intel AI examples (if attending afternoon)
  `$ git clone https://github.com/oneapi-src/oneAPI-samples.git`

Reaching a Compute Node

- Execute slurm command to get a compute node

```
$ sbatch intel-jupyterlab-tunnel.slurm
```

- View the job output file

```
$ cat intel-jupyterlab.job.*
```

- Copy, paste, and execute the ssh command that appears near the top of the output file. Example:

```
$ ssh -4 -L <port>:localhost:<port> <nodename>
```

Open Jupyter Lab in Browser

- Towards the end of the job output file (viewed like this)

  ```
  $ cat intel-jupyterlab.job.*
  ```

- login instructions will appear. Example:

```
To access the server, open this file in a browser:
    file:///home/<username>/.local/share/jupyter/runtime/jpserver-462321-open.html
Or copy and paste one of these URLs:
    http://localhost:<port>/lab?token=67b0e1263053b6bc449c59999984bbfc30a97fa61fcd9e18
 or http://127.0.0.1:<port>/lab?token=612b6b0iiic840c449c5a97fa61bbfc3fcd9e7b630530e18
```

- Copy and paste the link into your browser.

Jupyter Lab Environment

Intel Software

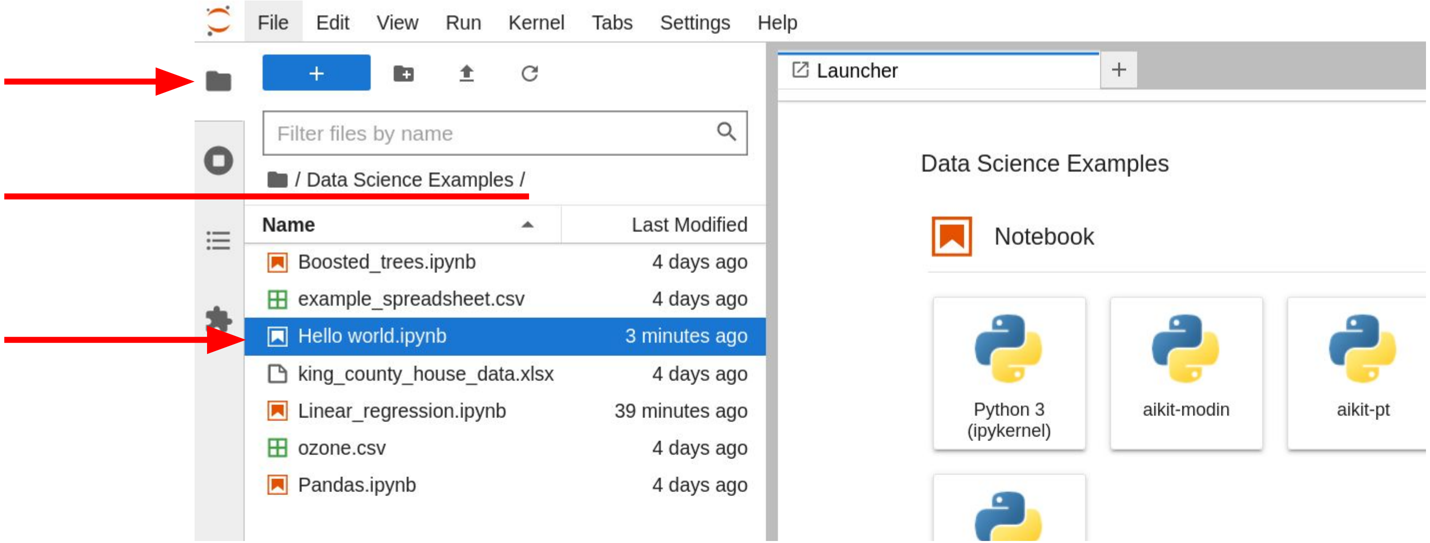
Intel Software integrated into HPRC Module Hierarchy

- `module load intel/Toolkits`
- (This command is in the slurm job file, already executed).

Provides access to a Conda environment where AI Toolkit and JupyterLab are installed.

Jupyter Lab File Navigator

Navigate to the “Hello_world.ipynb” file. Open by double-clicking.



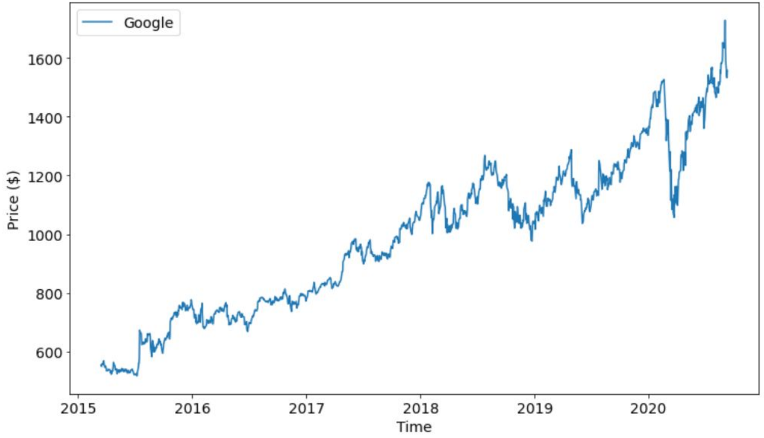
Jupyter Exercises

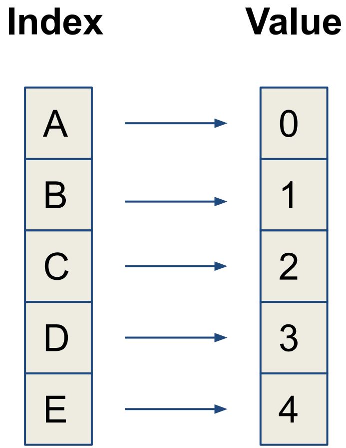
Complete the exercises in the Hello_World.ipynb notebook.

Data Structure with Pandas

Pandas Series

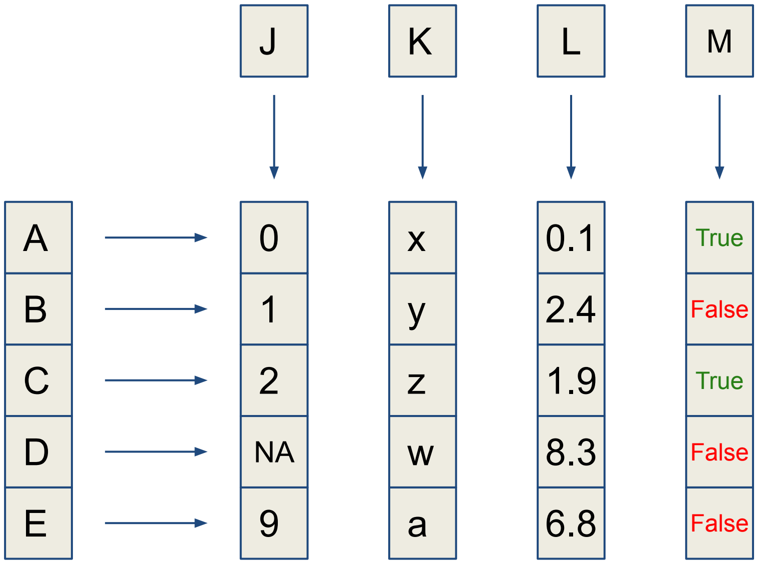
- One-dimensional labeled array
- Capable of holding any data type (integers, strings, floating point numbers, etc.)
- Example: time-series stock price data





Pandas DataFrame

- Primary Pandas data structure
- Like a dictionary of Series objects
- Tabular data structure
- Two-dimensional
- Size-mutable
- Heterogeneous



DataFrame Example

House sales data, King County

A	B	C	D	E	F	G	H
id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors
7129300520	20141013T0(	221900	3	1	1180	5650	1
6414100192	20141209T0(	538000	3	2.25	2570	7242	2
5631500400	20150225T0(	180000	2	1	770	10000	1
2487200875	20141209T0(	604000	4	3	1960	5000	1
1954400510	20150218T0(	510000	3	2	1680	8080	1
7237550310	20140512T0(	1.23E+06	4	4.5	5420	101930	1
1321400060	20140627T0(	257500	3	2.25	1715	6819	2
2008000270	20150115T0(	291850	3	1.5	1060	9711	1
2414600126	20150415T0(	229500	3	1	1780	7470	1

Pandas Exercises

Complete the exercises in the Pandas.ipynb notebook.

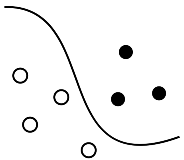
Machine Learning with Scikit Learn

Features of Scikit Learn

Classification

Identifying category of an object

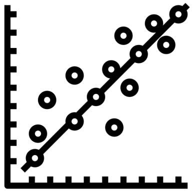
Applications: Spam detection, image recognition. **Algorithms**: SVM, nearest neighbors, random forest, and more...



Regression

Predicting a attribute for an object

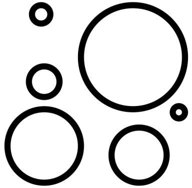
Applications: Drug response, Stock prices. **Algorithms**: SVR, nearest neighbors, random forest, and more...



Clustering

Grouping similar objects into sets

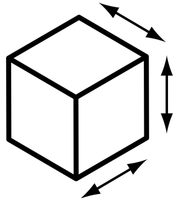
Applications: Customer segmentation, Grouping experiment outcomes **Algorithms:** k-Means, spectral clustering, mean-shift, and more...



Dimension Reduction

Reducing the number of dimensions

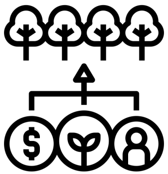
Applications: Visualization, Increased efficiency **Algorithms:** k-Means, feature selection, non-negative matrix factorization, and more...



Model Selection

Selecting models with parameter search

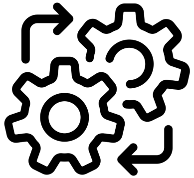
Applications: Improved accuracy via parameter tuning **Algorithms:** grid search, cross validation, metrics, and more...



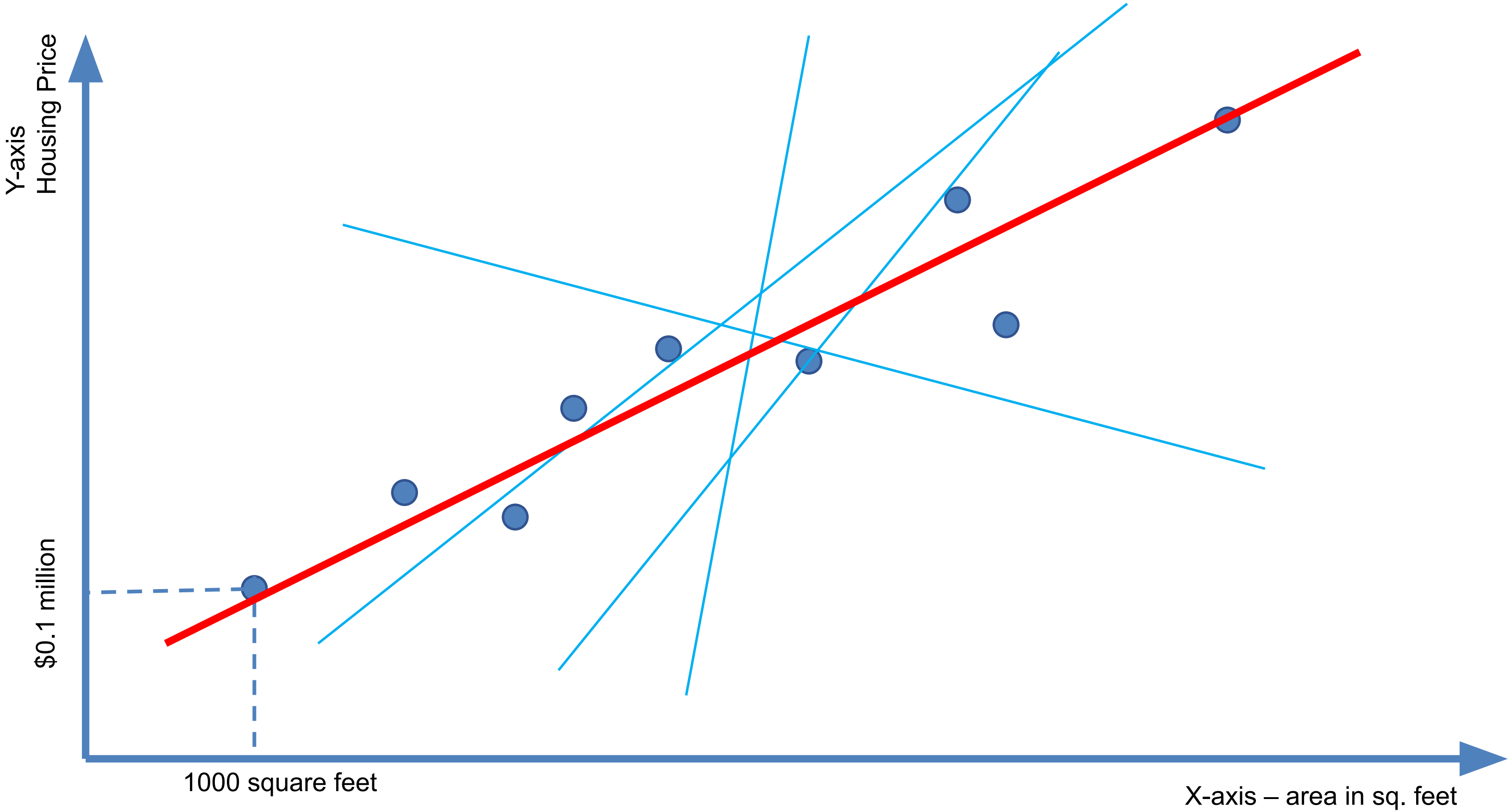
Preprocessing

Preprocessing data to prepare for modeling

Applications: Transforming input data such as text for use with machine learning algorithms. **Algorithms:** preprocessing, feature extraction, and more...

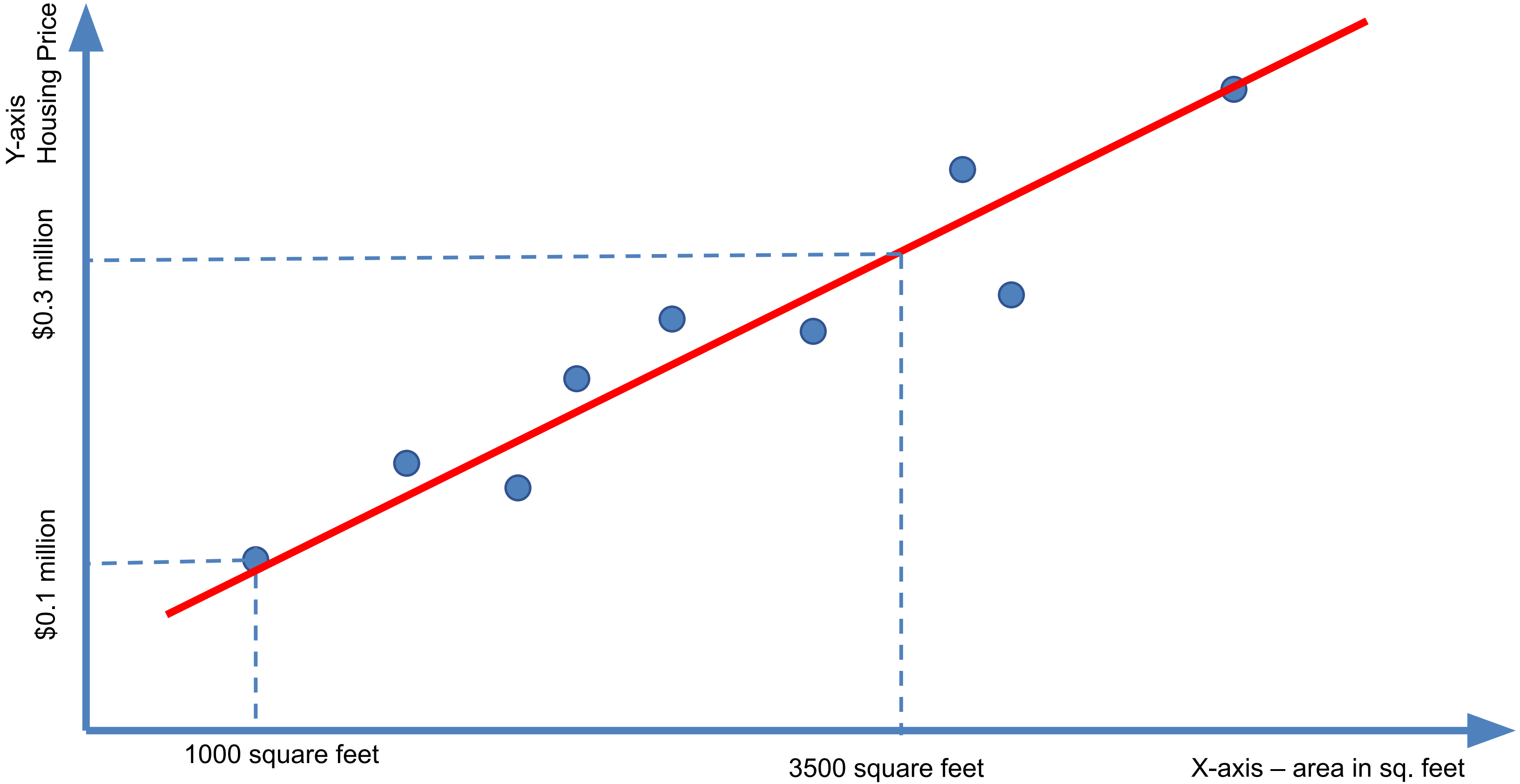


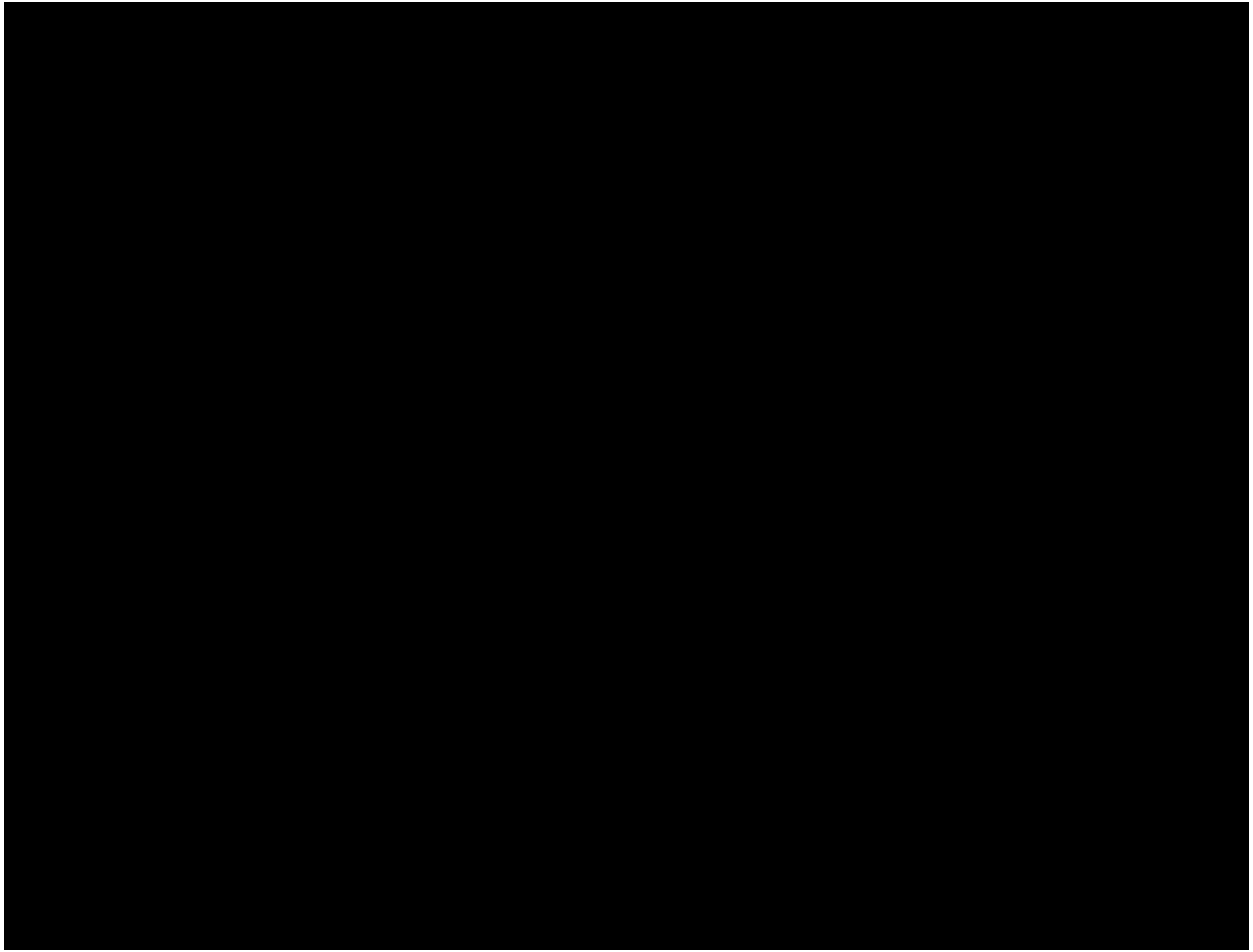
Regression



Y-axis
Housing Price
$0.1 million
1000 square feet
X-axis – area in sq. feet

Regression


Y-axis
Housing Price
$0.3 million
$0.1 million
1000 square feet
3500 square feet
X-axis – area in sq. feet



Scikit Learn Exercises

Complete the exercises in the Linear_regression.ipynb notebook.

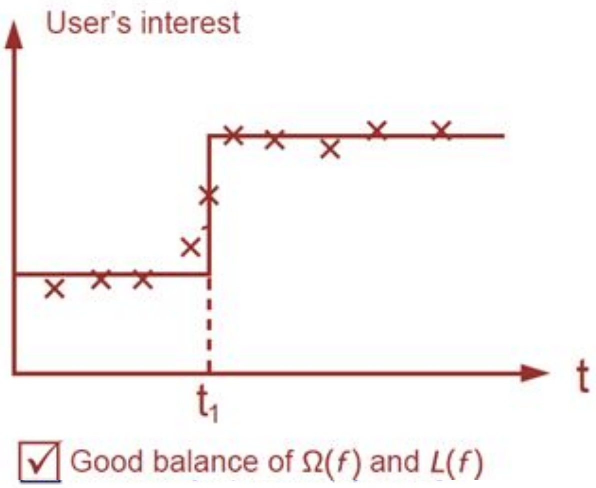
Machine Learning with XGBoost

Decision-making

- Prediction function is **step-wise**

$$f = \begin{cases} f_1 : & a<x<b \\ f_2 : & b<x<c \\ \ldots \end{cases}$$

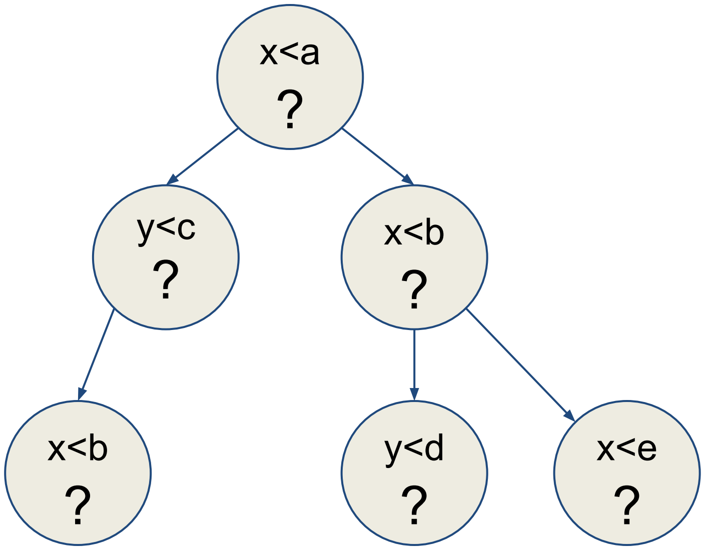
- Objective Function is **dual**
  - $obj(f)=L(f)+\Omega(f)$
  - L is prediction error
  - Ω is regularization



Images from https://xgboost.readthedocs.io/en/stable/tutorials/model.html

Decision Trees

- Complex Question?
  - Multiple Variables
  - Multiple Splits per Variable
  - Many Possible Tree Graphs
- “Learning” means growing the tree one Variable Split at a time



XGBoost Exercises

Complete the exercises in the Boosted_trees.ipynb notebook.

Shutdown JupyerLab

- In Browser
  - File → Shutdown → Yes
- Command line

```
$ squeue -u <username>
$ scancel <jobid>
```

Thank you

Contact: help@hprc.tamu.edu