Lab 3

Linear Regression & Logistic Regression

Q1a. Linear Regression: Used Car Dealership

- Imagine you work for a large online used car dealership and your boss would like you to estimate the price y (in dollars) the dealer should charge for a car based on the following features: x_1= car manufacturer, x_2= model, x_3= distance driven in miles, x_4= age in years, and binary features x_5= has side airbags, x_6= has leather seats, etc. For example, a feature vector for the i^{th} car could be $x^{(i)} = [4, 8, 17000, 5, 1, 0, \ldots]$ where manufacturer and model are represented as integers. You have collected data points from previous car sales, $(x^{(i)}, y^{(i)}),\ i = 1, \ldots, m$.
- You decide to use a linear regression model, $y = \sum_{j=0}^{n} \theta_j x_j$. In what circumstances should you choose gradient descent *vs.* normal equations to fit the parameters?
- Use gradient descent when the number of features n is large, i.e. the matrix (X^T)X is too large to invert (O(n3)), otherwise use normal equations.

Q1b. Linear Regression: Used Car Dealership

- Imagine you work for a large online used car dealership and your boss would like you to estimate the price y (in dollars) the dealer should charge for a car based on the following features: x_1= car manufacturer, x_2= model, x_3= distance driven in miles, x_4= age in years, and binary features x_5= has side airbags, x_6= has leather seats, etc. For example, a feature vector for the i^{th} car could be $x^{(i)} = [4, 8, 17000, 5, 1, 0, \dots]$ where manufacturer and model are represented as integers. You have collected data points from previous car sales, $(x^{(i)}, y^{(i)}),\ i = 1, \dots, m$.
- Suppose you use gradient descent. How can you tell if it is converging?
- By plotting the cost as a function of the number of iterations: convergence is likely when the decrease in cost diminishes.

Q1c. Linear Regression: Used Car Dealership

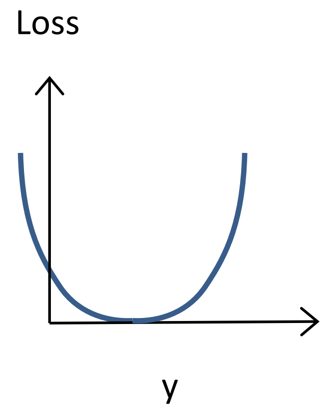
- Imagine you work for a large online used car dealership and your boss would like you to estimate the price y (in dollars) the dealer should charge for a car based on the following features: x_1= car manufacturer, x_2= model, x_3= distance driven in miles, x_4= age in years, and binary features x_5= has side airbags, x_6= has leather seats, etc. For example, a feature vector for the i^{th} car could be $x^{(i)} = [4, 8, 17000, 5, 1, 0, \ldots]$ where manufacturer and model are represented as integers. You have collected data points from previous car sales, $(x^{(i)}, y^{(i)}),\ i = 1, \ldots, m$.
- You find that your test accuracy is low. Name two things you can try to improve the result of linear regression without collecting any additional features.
- 1) Try a larger value for the step size;
- 2) In this case, some of the features have larger scale which could cause slow convergence, so we could use feature re-scaling to normalize all features, e.g. to [-1 1]

Q1d. Linear Regression: Used Car Dealership

- Imagine you work for a large online used car dealership and your boss would like you to estimate the price y (in dollars) the dealer should charge for a car based on the following features: x_1= car manufacturer, x_2= model, x_3= distance driven in miles, x_4= age in years, and binary features x_5= has side airbags, x_6= has leather seats, etc. For example, a feature vector for the i^{th} car could be $x^{(i)} = [4, 8, 17000, 5, 1, 0, \dots]$ where manufacturer and model are represented as integers. You have collected data points from previous car sales, $(x^{(i)}, y^{(i)}),\ i = 1, \dots, m$.
- You decide to add new features to improve your predictor. Is it a good idea to add distance driven in kilometers? Why or why not?
- No, because it is redundant with (a constant multiple of) the distance in miles and would not add new information.

Q1e. Linear Regression: Used Car Dealership

- Imagine you work for a large online used car dealership and your boss would like you to estimate the price y (in dollars) the dealer should charge for a car based on the following features: x_1= car manufacturer, x_2= model, x_3= distance driven in miles, x_4= age in years, and binary features x_5= has side airbags, x_6= has leather seats, etc. For example, a feature vector for the i^{th} car could be $x^{(i)} = [4, 8, 17000, 5, 1, 0, \ldots]$ where manufacturer and model are represented as integers. You have collected data points from previous car sales, $(x^{(i)}, y^{(i)}),\ i = 1, \ldots, m$.
- Typically in regression we minimize a square loss function, shown below. Does it make sense in this case? Why or why not?
- Yes, it makes sense, because it penalizes predicted values that are either below or above the actual sale price of the training example. However, as the next question suggests, there may be a better loss function we could use.



Q1f. Linear Regression: Used Car Dealership

- Imagine you work for a large online used car dealership and your boss would like you to estimate the price y (in dollars) the dealer should charge for a car based on the following features: x_1= car manufacturer, x_2= model, x_3= distance driven in miles, x_4= age in years, and binary features x_5= has side airbags, x_6= has leather seats, etc. For example, a feature vector for the i^{th} car could be $x^{(i)} = [4, 8, 17000, 5, 1, 0, \ldots]$ where manufacturer and model are represented as integers. You have collected data points from previous car sales, $(x^{(i)}, y^{(i)}),\ i = 1, \ldots, m$.
- Suppose you trained your model and it predicted a very low price for a particular Honda, $139. You check your training data, and find that all prices in the training examples are reasonable. The input features also look reasonable. What could be the reason for such a low prediction? How could you address it?
- Overfitting, which could be addressed by adding regularization to the model.

Q2a. Short Questions – True or False

- Suppose you want to predict if an email attachment contains a computer virus. What supervised machine learning method(s) would you use?
- Answer: classification

Q2b. Short Questions – True or False

- Suppose we use polynomial features for linear regression, then the hypothesis is linear in the original features [T/F]
- Answer: false, it is linear in the new polynomial features

Q2c. Short Questions – True or False

- The gradient descent update for logistic regression is identical to linear regression [T/F]
- Answer: false, they look similar but the hypothesis is different

Q3a. General Machine Learning Concepts

- Suppose you want to use training data D to adjust the parameters w of a model where $\mathrm{L(D) = p(D; w)}$ is the likelihood of the data. You want to prevent overfitting using a squared norm regularizer. What should your objective function look like? Should you minimize or maximize it?
- Answer: minimize the following objective (note, this maximizes $L(D)$):
- $J = -L(D) + \lambda ||w||^2$ or, $J = -\lambda L(D) + ||w||^2$
- where λ is a hyperparameter, or, equivalently, maximize $-J$.

Q3b. General Machine Learning Concepts

- What is cross-validation?
- Answer: When we split training data into training and extra validation set; learn model parameters on the training set, test and tune hyper-parameters on the validation set. We can do this multiple times, taking a different portion of original training data for the validation set each time (known as N-fold cross-validation).

Q3c. General Machine Learning Concepts

- How can we use it to prevent overfitting? Explain the procedure using the setup of (d).
- Answer: Train several models using different values of λ on the training set, test them on the validation set, and pick best model. This will reduce overfitting compared to tuning λ on training data.

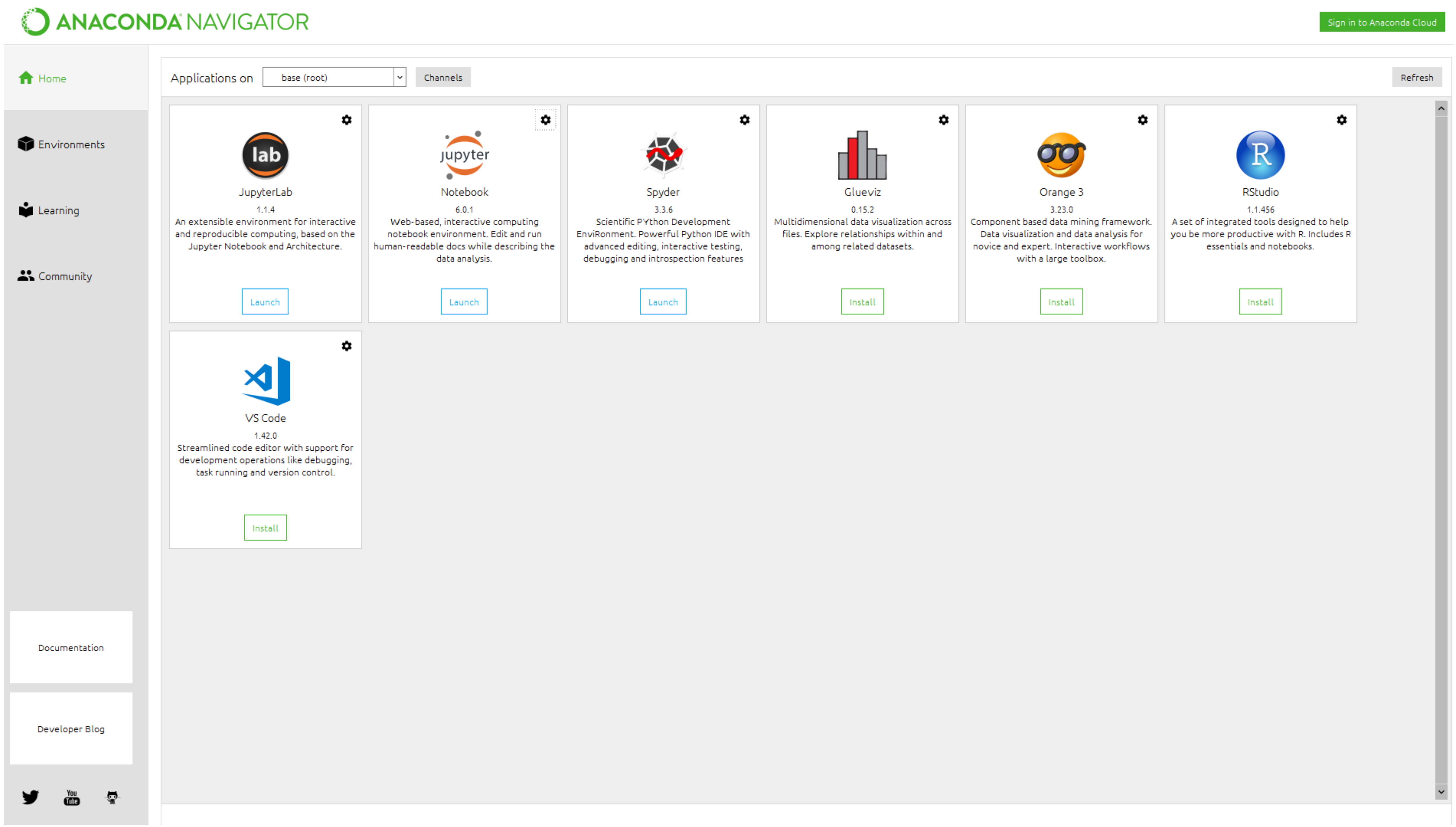
Anaconda Installation

- To run and solve assignments in this course, one must have a working IPython Notebook installation.

- The easiest way to set it up for both Windows and Linux is to:
  - install Anaconda: https://www.anaconda.com/distribution/ (Python Version 3)
  - save and run this file to your computer

- If you are new to Python or its scientific library, Numpy, there are some nice Tutorials: https://www.learnpython.org and http://scipy-lectures.org

- In Windows after installation, search "Anaconda Navigator". In the GUI menu, you can launch Jupyer Notebook or Jupyter Lab. These IDEs are based on a web-browser, you can enter localhost:8888 in a web browser and enter the work directory.


ANACONDA NAVIGATOR
Sign in to Anaconda Cloud
Home
Environments
Learning
Community
Applications on
base (root)
Channels
Refresh
JupyterLab
1.1.4
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.
Launch
Notebook
6.0.1
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
Launch
Spyder
3.3.6
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features
Launch
Glueviz
0.15.2
Multidimensional data visualization across files. Explore relationships within and among related datasets.
Install
Orange 3
3.23.0
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.
Install
RStudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.
Install
VS Code
1.42.0
Streamlined code editor with support for development operations like debugging, task running and version control.
Install
Documentation
Developer Blog