Intro to Machine Learning

In this lecture

- We begin with a high-level overview of machine learning.
- After the lecture, you should be able to recognize the different types of machine learning and how to evaluate different models.

Types of Machine Learning

- Supervised Learning:
  - Given independent data X and dependent data y, the goal is to predict y from X, that is, find a function f such that $f(X) \approx y$.
  - If y takes categorical values $\Rightarrow$ **Classification.**
  - If y takes numerical values $\Rightarrow$ **Regression.**
- Unsupervised Learning:
  - Only X is given and the task is to find some subgroups/structure.

Types of Machine Learning (cont.)

- Semi-supervised learning:
  - Similar to supervised learning, except that maybe not all observations in X have corresponding values y.
- Reinforcement learning:
  - Choose an action in a dynamic environment. This choice of action determines both a reward and the new state in the environment.

Methodology

- **Representation:**
    - This is the hypothesis space. How do we represent our model?
    - Choosing the representation determines the complexity/hypothesis space.
- **Evaluation:**
- We choose a criteria to say if a given model is good or not.
- **Optimization:**
    - The way we search for a good model on the hypothesis space.

Example: Linear regression

Let n denote the number of observations in X (number of rows).

- **Representation:** We assume there exists θ such that $f_\theta(X) = X\theta$ if a good representation of f (the "true" solution).
- **Evaluation:** For a given θ, we calculate

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i)^2$$

- **Optimization:** Update θ by gradient descent

$$\theta \leftarrow \theta - \alpha \nabla J(\theta)$$

Model Selection

Choosing among hypothesis spaces

- Besides linear functions, we could use polynomials. In particular, a polynomial of degree $n+1$ which would have zero error!
- However, the main goal of a machine learning model is to *generalize*.
- This means that the task should work well on *unseen* data.

Generalization

- To achieve generalization, we pretend we forget some of our data.
- While we use the majority for training our model, we leave a fraction aside for testing.

Types of Cross Validation

- **Holdout:**
  - Leave a fraction of the data aside, train the model on the rest.
- **Kfold**
  - Divide data in k parts, train in (k 1) of them and test in the other. Average the errors.
  - Use $k = 5$ for quick prototyping, k 10 for production/publication.
- **Leave one out:**
  - Train the model on all the data except for one observation.

Best practice: Do Holdout and K-fold

Learning Curve

- To see if our model is good, and to help us debug, we use a technique called **learning curve**.
- We train models with an increasing amount of data, and calculate the error.
- Then we plot of the number of observations against the model error.

Interpreting the learning curve

- **High test error, low train error:**
  - *Overfitting*. Simplify your model or add cross validation.
- **Big gap in train and test error:**
  - Problem with the data. Add more data, or check that the data in train and test comes from same distribution.
- **High value of train error:**
  - Likely a *bias* problem, use a more complicated model.

In summary

- Machine learning consists largely of trial and error across different representations for our problem.
- To ensure the results are reliable and will work well in unseen data, we use cross validation.
- To debug our models and decide whether we need better/more data or better algorithms, we use learning curves.