Linear regression unit:

- Simple linear regression with ordinary least squares (**today**)
- Lab activity: Linear regression (Wednesday)
- Deriving a closed form solution (Friday)
- Newton's method and gradient descent (next week Monday)
- Training linear regression using gradient descent (next week Wednesday)

Today:

- Foundational ideas
  - Problem statement for linear regression
  - Error, loss, and risk
  - Partial derivatives and gradients
- Deriving ordinary least squares for simple linear regression
- Variations: Multiple regression and regularization

Which of the following is *not* a hyperparameter of a k nearest neighbor classifier?

The number of neighbors
The distance metric
The curse of dimensionality

Which of the following is *not* true about k nearest neighbors classification?

It is non-parametric
The classifier stores all the training data

It works very well on many applications
The curse of dimensionality doesn't apply

Which if the following is *not* a vector distance metric?

Euclidean
Manhattan
Mahalanobis

Canberra
Curse of dimensionality

Typical sequence for a topic in this course:

- Problem statement and general concepts
- Concrete illustration and applications in lab
- Deep dive into the math
- Algorithmic details

Plan for this topics:

- Simple linear regression with ordinary least squares (**today**)
- Lab activity: Linear regression (Wednesday)
- Deriving a closed form solution (Friday)
- Newton's method and gradient descent (next week Monday)
- Training linear regression using gradient descent (next week Wednesday)

Linear regression gives us the opportunity to introduce larger ideas in machine learning:

- Training as finding parameters/weights
- Error vs loss (and cost and risk)
- Finding parameter using closed forms, as opposed to
- Finding parameters using iterative methods, especially the *gradient descent* algorithm

General form of the regression problem:

> *Given data* $\mathbf{X}$ *(N observations in D dimensions) and N target values as* $\vec{y}$*, find a function for predicting the values of new data points.*

- A **cost** function is a variable, formula, or function to be minimized in an optimization problem.
- The **error** of a model is the difference between the correct value and the computed value.
- A **loss** function is a measure of how well the model performs, usually applied to training data. Loss is an interpretation of error. Loss usually is treated as a function of the model.
- **Risk** is a measurement of how well the model performs on all possible data. Risk is the expected value of the loss function applied to arbitrary data, not just training data. The loss function applied to the training data is an experimental estimate of risk, that is, **empirical risk**.
- The **empirical risk minimization (ERM) framework** is the general strategy of finding a model that minimizes loss on the training data.

Polynomial regression:

$$y(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \cdots \theta_D x^D$$

Multiple regression:

$$y(\boldsymbol{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_D x_D = \theta_0 + \boldsymbol{\theta}^T \boldsymbol{x}$$

If we extend each observation so that it has 1 in position 0, that is $\boldsymbol{x} = [1, x_1, x_2, \ldots x_D]$ (so each observation acts like a vector of length $D + 1$), and interpret $\boldsymbol{\theta}$ as $[\theta_0, \theta_1, \theta_2, \ldots \theta_D]$, then the model family is

$$y(\boldsymbol{x}) = \boldsymbol{\theta}^T \boldsymbol{x}$$

Most general form, linear regression on arbitrary basis functions:

$$y(\boldsymbol{x}) = \theta_0 + \theta_1 \phi_1(\boldsymbol{x}) + \cdots \phi_D(\boldsymbol{x})$$

Loss function for ridge regularization (ridge regression):

$$\mathcal{L}_{ridge}(\boldsymbol{\theta}) = \underbrace{||\boldsymbol{y}^T - \boldsymbol{\theta}^T \mathbf{X}||^2}_{\text{original loss}} + \underbrace{\alpha||\boldsymbol{\theta}||^2}_{\text{regularizer}}$$

Loss function for LASSO regularization

$$\mathcal{L}_{LASSO}(\boldsymbol{\theta}) = ||\boldsymbol{y}^T - \boldsymbol{\theta}^T \mathbf{X}||^2 + \alpha \sum_{i=1}^{D} |\theta_i| = ||\boldsymbol{y}^T - \boldsymbol{\theta}^T \mathbf{X}||^2 + \alpha||\boldsymbol{\theta}||^1$$

Coming up:

Due Wed, Jan 29:
Read and respond to article about the Boston Housing Dataset
(Find pdf on Canvas.)

Due Thurs, Jan 30:
Read the textbook from Chapters 1 and 5 (see Canvas for specific sections)

Due Fri, Jan 31:
Do KNN programming assignment

Due Mon, Feb 3:
Propose project topic