Topics

- Beyond linear regression models
  - additive regression models, examples
  - generalization and cross-validation
  - population minimizer

- Statistical regression models
  - model formulation, motivation
  - maximum likelihood estimation
Linear regression

- Linear regression functions,

\[ f : \mathbb{R} \rightarrow \mathbb{R} \quad f(x; w) = w_0 + w_1 x, \quad \text{or} \]

\[ f : \mathbb{R}^d \rightarrow \mathbb{R} \quad f(x; w) = w_0 + w_1 x_1 + \ldots + w_d x_d \]

combined with the squared loss, are convenient because they are *linear in the parameters*. 
Linear regression

- Linear regression functions,

\[ f : \mathbb{R} \rightarrow \mathbb{R} \quad f(x; w) = w_0 + w_1 x, \text{ or} \]

\[ f : \mathbb{R}^d \rightarrow \mathbb{R} \quad f(x; w) = w_0 + w_1 x_1 + \ldots + w_d x_d \]

combined with the squared loss, are convenient because they are linear in the parameters.

- we get closed form estimates of the parameters

\[ \hat{w} = (X^T X)^{-1} X^T y \]

where, for example, \( y = [y_1, \ldots, y_n]^T \).
Linear regression

- Linear regression functions,
  \[ f : \mathbb{R} \rightarrow \mathbb{R} \quad f(x; w) = w_0 + w_1 x, \quad \text{or} \]
  \[ f : \mathbb{R}^d \rightarrow \mathbb{R} \quad f(x; w) = w_0 + w_1 x_1 + \ldots + w_d x_d \]

combined with the squared loss, are convenient because they are linear in the parameters.

- we get closed form estimates of the parameters
  \[ \hat{w} = (X^T X)^{-1} X^T y \]
  where, for example, \( y = [y_1, \ldots, y_n]^T \).

- the resulting prediction errors \( \epsilon_i = y_i - f(x_i; \hat{w}) \) are uncorrelated with any linear function of the inputs \( x \).
Linear regression

- Linear regression functions,
  \[ f : \mathcal{R} \rightarrow \mathcal{R} \quad f(x; w) = w_0 + w_1 x, \quad \text{or} \]
  \[ f : \mathcal{R}^d \rightarrow \mathcal{R} \quad f(x; w) = w_0 + w_1 x_1 + \ldots + w_d x_d \]

  combined with the squared loss, are convenient because they are linear in the parameters.

- we get closed form estimates of the parameters
  \[ \hat{w} = (X^T X)^{-1} X^T y \]
  where, for example, \( y = [y_1, \ldots, y_n]^T \).

- the resulting prediction errors \( \epsilon_i = y_i - f(x_i; \hat{w}) \) are uncorrelated with any linear function of the inputs \( x \).

- we can easily extend these to non-linear functions of the inputs while still keeping them linear in the parameters.
Beyond linear regression

- Example extension: $m^{th}$ order polynomial regression where $f : \mathcal{R} \rightarrow \mathcal{R}$ is given by

$$f(x; \mathbf{w}) = w_0 + w_1 x + \ldots + w_{m-1} x^{m-1} + w_m x^m$$

- linear in the parameters, non-linear in the inputs
- solution as before

$$\hat{\mathbf{w}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

where

$$\hat{\mathbf{w}} = \begin{bmatrix} \hat{w}_0 \\ \hat{w}_1 \\ \vdots \\ \hat{w}_m \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^m \\ 1 & x_2 & x_2^2 & \ldots & x_2^m \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ 1 & x_n & x_n^2 & \ldots & x_n^m \end{bmatrix}$$
Polynomial regression

- Degree = 1
- Degree = 3
- Degree = 5
- Degree = 7
Complexity and overfitting

- With limited training examples our polynomial regression model may achieve zero training error but nevertheless has a large test (generalization) error

$$\text{train} \quad \frac{1}{n} \sum_{t=1}^{n} (y_t - f(x_t; \hat{w}))^2 \approx 0$$

$$E_{(x,y) \sim P} (y - f(x; \hat{w}))^2 \gg 0$$

- We suffer from over-fitting when the training error no longer bears any relation to the generalization error
Avoiding over-fitting: cross-validation

- Cross-validation allows us to estimate the generalization error based on training examples alone.

Leave-one-out cross-validation treats each training example in turn as a test example:

\[
CV = \frac{1}{n} \sum_{i=1}^{n} \left( y_i - f(x_i; \hat{w}^{-i}) \right)^2
\]

where \( \hat{w}^{-i} \) are the least squares estimates of the parameters without the \( i^{th} \) training example.
Polynomial regression: example cont’d

degree = 1, CV = 0.6

degree = 3, CV = 1.5

degree = 5, CV = 6.0

degree = 7, CV = 15.6
Additive models

- More generally, predictions can be based on a linear combination of a set of basis functions (or features) \( \{\phi_1(x), \ldots, \phi_m(x)\} \), where each \( \phi_i(x) : \mathbb{R}^d \rightarrow \mathbb{R} \), and

\[
f(x; \mathbf{w}) = w_0 + w_1 \phi_1(x) + \ldots + w_m \phi_m(x)
\]

- Examples:

  If \( \phi_i(x) = x^i \), \( i = 1, \ldots, m \), then

\[
f(x; \mathbf{w}) = w_0 + w_1 x + \ldots + w_{m-1} x^{m-1} + w_m x^m
\]
Additive models

- More generally, predictions can be based on a linear combination of a set of basis functions (or features) \( \{\phi_1(x), \ldots, \phi_m(x)\} \), where each \( \phi_i(x) : \mathcal{R}^d \to \mathcal{R} \), and

\[
f(x; w) = w_0 + w_1\phi_1(x) + \ldots + w_m\phi_m(x)
\]

- Examples:

  If \( \phi_i(x) = x^i \), \( i = 1, \ldots, m \), then

\[
f(x; w) = w_0 + w_1x + \ldots + w_{m-1}x^{m-1} + w_mx^m
\]

  If \( m = d \), \( \phi_i(x) = x_i \), \( i = 1, \ldots, d \), then

\[
f(x; w) = w_0 + w_1x_1 + \ldots + w_dx_d
\]
Additive models cont’d

- The basis functions can capture various (e.g., qualitative) properties of the inputs.

For example: we can try to rate companies based on text descriptions

$$x = \text{text document (collection of words)}$$

$$\phi_i(x) = \begin{cases} 1 & \text{if word } i \text{ appears in the document} \\ 0 & \text{otherwise} \end{cases}$$

$$f(x; w) = w_0 + \sum_{i \in \text{words}} w_i \phi_i(x)$$
Additive models cont’d

- We can also make predictions by gauging the similarity of examples to “prototypes”.

For example, our additive regression function could be

\[ f(x; w) = w_0 + w_1\phi_1(x) + \ldots + w_m\phi_m(x) \]

where the basis functions are “radial basis functions”

\[ \phi_k(x) = \exp\left\{ -\frac{1}{2\sigma^2}\|x - x_k\|^2 \right\} \]

measuring the similarity to the prototypes; \( \sigma^2 \) controls how quickly the basis function vanishes as a function of the distance to the prototype.

(training examples themselves could serve as prototypes)
Additive models cont’d

• We can view the additive models graphically in terms of simple “units” and “weights”

\[ f(x; w) = \sum_{i=1}^{m} w_i \phi_i(x) \]

• In *neural networks* the basis functions themselves have adjustable parameters (cf. prototypes)
Squared loss and population minimizer

- What do we get if we have unlimited training examples (the whole population) and no constraints on the regression function?

\[
\text{minimize } E_{(x,y) \sim P} (y - f(x))^2
\]

with respect to an unconstrained function \( f : \mathcal{R} \rightarrow \mathcal{R} \)
Squared loss and population minimizer

- To minimize

\[
E_{(x,y) \sim P} (y - f(x))^2 = E_{x \sim P_x} \left[ E_{y \sim P_{y|x}} (y - f(x))^2 \right]
\]

we can focus on each \(x\) separately since \(f(x)\) can be chosen independently for each different \(x\). For any particular \(x\) we can

\[
\frac{\partial}{\partial f(x)} E_{y \sim P_{y|x}} (y - f(x))^2 = 2E_{y \sim P_{y|x}} (y - f(x)) = 2(E\{y|x\} - f(x)) = 0
\]

Thus the function we are trying to approximate is the conditional expectation

\[
f^*(x) = E\{y|x\}
\]
Topics

• Beyond linear regression models
  – additive regression models, examples
  – generalization and cross-validation
  – population minimizer

• Statistical regression models
  – model formulation, motivation
  – maximum likelihood estimation
Statistical view of linear regression

- In a statistical regression model we model both the function and noise

\[
\text{Observed output} = \text{function + noise}
\]

\[
y = f(x; w) + \epsilon
\]

where, e.g., \( \epsilon \sim N(0, \sigma^2) \).

- Whatever we cannot capture with our chosen family of functions will be interpreted as noise
Statistical view of linear regression

• $f(x; w)$ is trying to capture the mean of the observations $y$ given the input $x$:

$$E\{ y \mid x \} = E\{ f(x; w) + \epsilon \mid x \} = f(x; w)$$

where $E\{ y \mid x \}$ is the conditional expectation of $y$ given $x$, evaluated according to the model (not according to the underlying distribution $P$)
Statistical view of linear regression

• According to our statistical model

\[ y = f(x; w) + \epsilon, \quad \epsilon \sim N(0, \sigma^2) \]

the outputs \( y \) given \( x \) are normally distributed with mean \( f(x; w) \) and variance \( \sigma^2 \):

\[
p(y|\mathbf{x}, \mathbf{w}, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{1}{2\sigma^2}(y - f(x; w))^2 \right\}
\]

(we model the uncertainty in the predictions, not just the mean)

• Loss function? Estimation?
Maximum likelihood estimation

- Given observations \( D_n = \{(x_1, y_1), \ldots, (x_n, y_n)\} \) we find the parameters \( w \) that maximize the (conditional) likelihood of the outputs

\[
L(D_n; w, \sigma^2) = \prod_{i=1}^{n} p(y_i|x_i, w, \sigma^2)
\]

Example: linear function

\[
p(y|x, w, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{1}{2\sigma^2}(y - w_0 - w_1x)^2 \right\}
\]

(why is this a bad fit according to the likelihood criterion?)
Maximum likelihood estimation cont’d

Likelihood of the observed outputs:

\[ L(D; w, \sigma^2) = \prod_{i=1}^{n} P(y_i|x_i, w, \sigma^2) \]

• It is often easier (but equivalent) to try to maximize the log-likelihood:

\[
\begin{align*}
    l(D; w, \sigma^2) &= \log L(D; w, \sigma^2) = \sum_{i=1}^{n} \log P(y_i|x_i, w, \sigma^2) \\
    &= \sum_{i=1}^{n} \left(-\frac{1}{2\sigma^2}(y_i - f(x_i; w))^2 - \log \sqrt{2\pi\sigma^2}\right) \\
    &= \left(-\frac{1}{2\sigma^2}\right) \sum_{i=1}^{n} (y_i - f(x_i; w))^2 + \ldots
\end{align*}
\]
Maximum likelihood estimation cont’d

• Maximizing log-likelihood is equivalent to minimizing empirical loss when the loss is defined according to

\[
\text{Loss}(y_i, f(x_i; w)) = -\log P(y_i|x_i, w, \sigma^2)
\]

Loss defined as the negative log-probability is known as the \textit{log-loss}.
Maximum likelihood estimation cont’d

- The log-likelihood of observations

\[
\log L(D; w, \sigma^2) = \sum_{i=1}^{n} \log P(y_i|\mathbf{x}_i, w, \sigma^2)
\]

is a generic fitting criterion and can be used to estimate the noise variance \(\sigma^2\) as well.

- Let \(\hat{w}\) be the maximum likelihood (here least squares) setting of the parameters. What is the maximum likelihood estimate of \(\sigma^2\), obtained by solving

\[
\frac{\partial}{\partial \sigma^2} \log L(D; w, \sigma^2) = 0
\]
Maximum likelihood estimation cont’d

• The log-likelihood of observations

\[
\log L(D; \mathbf{w}, \sigma^2) = \sum_{i=1}^{n} \log P(y_i | x_i, \mathbf{w}, \sigma^2)
\]

is a generic fitting criterion and can be used to estimate the noise variance \( \sigma^2 \) as well.

• Let \( \hat{\mathbf{w}} \) be the maximum likelihood (here least squares) setting of the parameters. The maximum likelihood estimate of the noise variance \( \sigma^2 \) is

\[
\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i; \hat{\mathbf{w}}))^2
\]

i.e., the mean squared prediction error.