LINEAR REGRESSION
Example: Old Faithful
Can we meaningfully predict the time between eruptions only using the duration of the last eruption?
Can we meaningfully predict the time between eruptions only using the duration of the last eruption?
**Example: Old Faithful**

One model for this

\[(\text{wait time}) \approx w_0 + (\text{last duration}) \times w_1\]

- $w_0$ and $w_1$ are to be learned.
- This is an example of linear regression.

**Refresher**

$w_1$ is the slope, $w_0$ is called the intercept, bias, shift, offset.
Two inputs

\[ (\text{output}) \approx w_0 + (\text{input 1}) \times w_1 + (\text{input 2}) \times w_2 \]

With two inputs the intuition is the same \[ y = w_0 + x_1w_1 + x_2w_2 \]
Regression: Problem Definition

Data
Input: \( x \in \mathbb{R}^d \) (i.e., measurements, covariates, features, indepen. variables)
Output: \( y \in \mathbb{R} \) (i.e., response, dependent variable)

Goal
Find a function \( f : \mathbb{R}^d \to \mathbb{R} \) such that \( y \approx f(x; w) \) for the data pair \( (x, y) \).
\( f(x; w) \) is called a regression function. Its free parameters are \( w \).

Definition of linear regression
A regression method is called linear if the prediction \( f \) is a linear function of the unknown parameters \( w \).
Model
The linear regression model we focus on now has the form

\[ y_i \approx f(x_i; w) = w_0 + \sum_{j=1}^{d} x_{ij} w_j. \]

Model learning
We have the set of training data \((x_1, y_1) \ldots (x_n, y_n)\). We want to use this data to learn a \(w\) such that \(y_i \approx f(x_i; w)\). But we first need an objective function to tell us what a “good” value of \(w\) is.

Least squares
The least squares objective tells us to pick the \(w\) that minimizes the sum of squared errors

\[ w_{LS} = \arg \min_w \sum_{i=1}^{n} (y_i - f(x_i; w))^2 \equiv \arg \min_w \mathcal{L}. \]
Observations:
Vertical length is error.

The objective function $\mathcal{L}$ is the sum of all the squared lengths.

Find weights $(w_1, w_2)$ plus an offset $w_0$ to minimize $\mathcal{L}$.

$(w_0, w_1, w_2)$ defines this plane.
Example: Education, Seniority and Income

2-dimensional problem

Input: (education, seniority) ∈ \( \mathbb{R}^2 \).

Output: (income) ∈ \( \mathbb{R} \)

Model: (income) \( \approx \) \( w_0 + (\text{education})w_1 + (\text{seniority})w_2 \)

Question: Both \( w_1, w_2 \) > 0. What does this tell us?

Answer: As education and/or seniority goes up, income tends to go up.

(Caveat: This is a statement about correlation, not causation.)
Thus far
We have data pairs \((x_i, y_i)\) of measurements \(x_i \in \mathbb{R}^d\) and a response \(y_i \in \mathbb{R}\). We believe there is a linear relationship between \(x_i\) and \(y_i\),

\[
y_i = w_0 + \sum_{j=1}^{d} x_{ij}w_j + \epsilon_i
\]

and we want to minimize the objective function

\[
\mathcal{L} = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - w_0 - \sum_{j=1}^{d} x_{ij}w_j)^2
\]

with respect to \((w_0, w_1, \ldots, w_d)\).

Can math notation make this easier to look at/work with?
We think of data with $d$ dimensions as a column vector:

$$x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{bmatrix} \quad \text{(e.g.)} \Rightarrow \begin{bmatrix} \text{age} \\ \text{height} \\ \vdots \\ \text{income} \end{bmatrix}$$

A set of $n$ vectors can be stacked into a matrix:

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1d} \\ x_{21} & \cdots & x_{2d} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nd} \end{bmatrix} = \begin{bmatrix} -x_1^T \\ -x_2^T \\ \vdots \\ -x_n^T \end{bmatrix}$$

Assumptions for now:

- All features are treated as continuous-valued ($x \in \mathbb{R}^d$)
- We have more observations than dimensions ($d < n$)
Usually, for linear regression (and classification) we include an intercept term $w_0$ that doesn’t interact with any element in the vector $x \in \mathbb{R}^d$.

It will be convenient to attach a 1 to the first dimension of each vector $x_i$ (which we indicate by $x_i \in \mathbb{R}^{d+1}$) and in the first column of the matrix $X$:

$$x_i = \begin{bmatrix} 1 \\ x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_{11} & \ldots & x_{1d} \\ 1 & x_{21} & \ldots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \ldots & x_{nd} \end{bmatrix} = \begin{bmatrix} 1 - x_1^T - \\ 1 - x_2^T - \\ \vdots \\ 1 - x_n^T - \end{bmatrix}.$$  

We also now view $w = [w_0, w_1, \ldots, w_d]^T$ as $w \in \mathbb{R}^{d+1}$. 


Least squares in vector form

Original least squares objective function: \( \mathcal{L} = \sum_{i=1}^{n} (y_i - w_0 - \sum_{j=1}^{d} x_{ij}w_j)^2 \)

Using vectors, this can now be written: \( \mathcal{L} = \sum_{i=1}^{n} (y_i - x_i^T w)^2 \)

Least squares solution (vector version)

We can find \( w \) by setting,

\[
\nabla_w \mathcal{L} = 0 \quad \Rightarrow \quad \sum_{i=1}^{n} \nabla_w (y_i^2 - 2w^T x_i y_i + w^T x_i x_i^T w) = 0.
\]

Solving gives,

\[
- \sum_{i=1}^{n} 2y_i x_i + \left( \sum_{i=1}^{n} 2x_i x_i^T \right) w = 0 \quad \Rightarrow \quad w_{LS} = \left( \sum_{i=1}^{n} x_i x_i^T \right)^{-1} \left( \sum_{i=1}^{n} y_i x_i \right).
\]
Least squares solution (matrix version)

Least squares in matrix form is even cleaner.

Start by organizing the $y_i$ in a column vector, $y = [y_1, \ldots, y_n]^T$. Then

$$\mathcal{L} = \sum_{i=1}^{n} (y_i - x_i^T w)^2 = \|y - Xw\|^2 = (y - Xw)^T (y - Xw).$$

If we take the gradient with respect to $w$, we find that

$$\nabla_w \mathcal{L} = 2X^T Xw - 2X^T y = 0 \quad \Rightarrow \quad w_{ls} = (X^T X)^{-1} X^T y.$$
Recall: Matrix $\times$ vector \( (X^T y = \sum_{i=1}^{n} y_i x_i) \)

$$
\begin{bmatrix}
  x_1 & x_2 & \cdots & x_n
\end{bmatrix}
\begin{bmatrix}
  y_1 \\
y_2 \\
  \vdots \\
y_n
\end{bmatrix}
= y_1 \begin{bmatrix}
  x_1 \\
\end{bmatrix}
+ y_2 \begin{bmatrix}
  x_2 \\
\end{bmatrix}
+ \cdots + y_n \begin{bmatrix}
  x_n \\
\end{bmatrix}
$$

Recall: Matrix $\times$ matrix \( (X^T X = \sum_{i=1}^{n} x_i x_i^T) \)

$$
\begin{bmatrix}
  x_1 & x_2 & \cdots & x_n
\end{bmatrix}
\begin{bmatrix}
  -x_1^T \\
  -x_2^T \\
  \vdots \\
  -x_n^T
\end{bmatrix}
= x_1 x_1^T + \cdots + x_n x_n^T.
$$
Two notations for the key equation

\[ w_{LS} = \left( \sum_{i=1}^{n} x_i x_i^T \right)^{-1} \left( \sum_{i=1}^{n} y_i x_i \right) \iff w_{LS} = (X^T X)^{-1} X^T y. \]

Making Predictions

We use \( w_{LS} \) to make predictions.

Given \( x_{\text{new}} \), the least squares prediction for \( y_{\text{new}} \) is

\[ y_{\text{new}} \approx x_{\text{new}}^T w_{LS} \]
Potential issues

Calculating \( w_{LS} = (X^T X)^{-1} X^T y \) assumes \((X^T X)^{-1}\) exists.

When doesn’t it exist?

Answer: When \(X^T X\) is not a full rank matrix.

When is \(X^T X\) full rank?

Answer: When the \(n \times (d + 1)\) matrix \(X\) has at least \(d + 1\) linearly independent rows. This means that any point in \(\mathbb{R}^{d+1}\) can be reached by a weighted combination of \(d + 1\) rows of \(X\).

Obviously if \(n < d + 1\), we can’t do least squares. If \((X^T X)^{-1}\) doesn’t exist, there are an infinite number of possible solutions.

Takeaway: We want \(n \gg d\) (i.e., \(X\) is “tall and skinny”).
BROADENING LINEAR REGRESSION
BROADENING LINEAR REGRESSION

\[ y = w_0 + w_1 x \]
Broadening linear regression

\[ y = w_0 + w_1 x + w_2 x^2 + w_3 x^3 \]
Recall: Definition of linear regression

A regression method is called linear if the prediction $f$ is a linear function of
the unknown parameters $w$.

- Therefore, a function such as $y = w_0 + w_1x + w_2x^2$ is linear in $w$.
The LS solution is the same, only the preprocessing is different.
- E.g., Let $(x_1, y_1) \ldots (x_n, y_n)$ be the data, $x \in \mathbb{R}$, $y \in \mathbb{R}$. For a $p$th-order polynomial approximation, construct the matrix

\[
X = \begin{bmatrix}
1 & x_1 & x_1^2 & \cdots & x_1^p \\
1 & x_2 & x_2^2 & \cdots & x_2^p \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & x_n & x_n^2 & \cdots & x_n^p 
\end{bmatrix}
\]

- Then solve exactly as before: $w_{LS} = (X^TX)^{-1}X^Ty$. 
Polynomial regression (Mth order)
POLYNOMIAL REGRESSION ($M$th ORDER)

\[ P(x) = \sum_{m=0}^{M-1} a_m x^m \]

$M = 1$
POLYNOMIAL REGRESSION ($M$th ORDER)

\[ M = 3 \]
Example: 2nd and 3rd order polynomial regression in $\mathbb{R}^2$

The width of $X$ grows as $(\text{order}) \times (\text{dimensions}) + 1$.

2nd order:

$$y_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i1}^2 + w_4 x_{i2}^2$$

3rd order:

$$y_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i1}^2 + w_4 x_{i2}^2 + w_5 x_{i1}^3 + w_6 x_{i2}^3$$
FURTHER EXTENSIONS

More generally, for $x_i \in \mathbb{R}^{d+1}$ least squares linear regression can be performed on functions $f(x_i; w)$ of the form

$$y_i \approx f(x_i, w) = \sum_{s=1}^{S} g_s(x_i)w_s.$$ 

For example,

$$g_s(x_i) = \begin{cases} 
  x_{ij}^2 & 
  
  \log x_{ij} & 
  \mathbb{I}(x_{ij} < a) & 
  \mathbb{I}(x_{ij} < x_{ij}')
\end{cases}$$

As long as the function is linear in $w_1, \ldots, w_S$, we can construct the matrix $X$ by putting the transformed $x_i$ on row $i$, and solve $w_{LS} = (X^T X)^{-1}X^T y$.

One caveat is that, as the number of functions increases, we need more data to avoid overfitting.
Thinking geometrically about least squares regression helps a lot.

- We want to minimize $\|y - Xw\|^2$. Think of the vector $y$ as a point in $\mathbb{R}^n$. We want to find $w$ in order to get the product $Xw$ close to $y$.

- If $X_j$ is the $j$th column of $X$, then $Xw = \sum_{j=1}^{d+1} w_j X_j$.

- That is, we weight the columns in $X$ by values in $w$ to approximate $y$.

- The LS solutions returns $w$ such that $Xw$ is as close to $y$ as possible in the Euclidean sense (i.e., intuitive “direct-line” distance).
**Geometry of Least Squares Regression**

\[
\text{arg min}_w \| y - Xw \|^2 \Rightarrow w_{LS} = (X^TX)^{-1}X^Ty.
\]

The columns of \( X \) define a \( d + 1 \)-dimensional subspace in the higher dimensional \( \mathbb{R}^n \).

The closest point in that subspace is the *orthonormal projection* of \( y \) into the *column space* of \( X \).

Right: \( y \in \mathbb{R}^3 \) and data \( x_i \in \mathbb{R} \).

\[
X_1 = [1, 1, 1]^T \text{ and } X_2 = [x_1, x_2, x_3]^T
\]

The approximation is \( \hat{y} = Xw_{LS} = X(X^TX)^{-1}X^Ty \).
GEOMETRY OF LEAST SQUARES REGRESSION

(a) $y_i \approx w_0 + x_i^T w$ for $i = 1, \ldots, n$

(b) $y \approx Xw$

There are some key differences between (a) and (b) worth highlighting as you try to develop the corresponding intuitions.

(a) Can be shown for all $n$, but only for $x_i \in \mathbb{R}^2$ (not counting the added 1).

(b) This corresponds to $n = 3$ and one-dimensional data: $X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \end{bmatrix}$. 