Chapter ML:III (continued)

III. Linear Models

- Logistic Regression
- Loss Computation
- Overfitting
- Regularization
Overfitting

**Definition 9 (Overfitting)**

Let $D$ be a set of examples and let $H$ be a hypothesis space. The hypothesis $h \in H$ is considered to overfit $D$ if an $h' \in H$ with the following property exists:

$$\text{Err}(h, D) < \text{Err}(h', D) \quad \text{and} \quad \text{Err}^*(h) > \text{Err}^*(h'),$$

where $\text{Err}^*(h)$ denotes the **true misclassification rate** of $h$, while $\text{Err}(h, D)$ denotes the error of $h$ on the example set $D$. 
Overfitting

**Definition 9 (Overfitting)**

Let $D$ be a set of examples and let $H$ be a hypothesis space. The hypothesis $h \in H$ is considered to overfit $D$ if an $h' \in H$ with the following property exists:

$$
\text{Err}(h, D) < \text{Err}(h', D) \quad \text{and} \quad \text{Err}^*(h) > \text{Err}^*(h'),
$$

where $\text{Err}^*(h)$ denotes the true misclassification rate of $h$, while $\text{Err}(h, D)$ denotes the error of $h$ on the example set $D$.

Reasons for overfitting are often rooted in the example set $D$:

- $D$ is noisy and we “learn noise”
- $D$ is biased and hence not representative
- $D$ is too small and hence pretends unrealistic data properties
Overfitting
Example: Linear Regression
Overfitting

Example: Linear Regression

\[ y(x) = w_0 + w_1 \cdot x \]
Overfitting

Example: Linear Regression

\[ y(x) = w_0 + w_1 \cdot x + w_2 \cdot x^2 \]

(basis expansion)
Overfitting
Example: Linear Regression

\[ y(x) = w_0 + \sum_{i=1}^{6} w_i \cdot x_i \]

(basis expansion)
Overfitting
Example: Linear Regression

Given three polynomial model functions $y(x) = w^T x$ of degrees 1, 2, and 6, and a training set of $D_{tr}$, select the function that best fits the data:

$$\text{RSS}(w) \gg 0$$

$$\text{RSS}(w) > 0$$

$$\text{RSS}(w) = 0$$
Overfitting
Example: Linear Regression

Given three polynomial model functions \( y(x) = w^T x \) of degrees 1, 2, and 6, and a training set of \( D_{tr} \), select the function that best fits the data:

\[
\begin{align*}
\text{RSS}(w) &> 0 \\
\text{RSS}(w) &> 0 \\
\text{RSS}(w) &= 0
\end{align*}
\]

Questions:
- Which model function (hypothesis) is the best choice?
- How to choose among the many hypothesis spaces available?
Overfitting

Example: Linear Regression

Given three polynomial model functions $y(x) = w^T x$ of degrees 1, 2, and 6, and a training set of $D_{tr}$, select the function that best fits the data:

$$\text{RSS}(w) \gg 0$$  
$$\text{RSS}(w) > 0$$  
$$\text{RSS}(w) \gg 0$$

Let $D_{ts}$ be a set of test examples.

If $D = D_{tr} \cup D_{ts}$ is representative of the real-world population in $X$, the quadratic model function $y(x) = w_0 + w_1 \cdot x + w_2 \cdot x^2$ is the closest match.
Overfitting
Estimation

Let $D_{tr} \subset D$ be the training set. Then $Err^*(h)$ can be estimated with a test set $D_{ts} \subset D$ where $D_{ts} \cap D_{tr} = \emptyset$ [holdout estimation]. The hypothesis $h \in H$ is considered to overfit $D$ if an $h' \in H$ with the following property exists:

$$Err(h, D_{tr}) < Err(h', D_{tr}) \quad \text{and} \quad Err(h, D_{ts}) > Err(h', D_{ts})$$
Overfitting
Mitigation Strategies

How to detect overfitting:

- Visual inspection
  Apply projection or embedding for dimensionalities $p > 3$.

- Validation
  Given a validation set, the difference $Err_{val}(y) - Err_{tr}(y)$ is too large.
Overfitting
Mitigation Strategies

How to detect overfitting:

- **Visual inspection**
  Apply projection or embedding for dimensionalities $p > 3$.
- **Validation**
  Given a validation set, the difference $Err_{val}(y) - Err_{tr}(y)$ is too large.

How to prevent overfitting:

- **Early stopping through model selection**
  Let $m$ be the number of training steps, and $y_{\pi_1}, \ldots, y_{\pi_m}$ the models obtained after each step.
  Indicator of overfitting: The difference $Err_{val}(y_{\pi_i}) - Err_{tr}(y_{\pi_i})$ increases with $i$.
- **Increasing quantity or quality of the training data $D$**
  Quantity: More data averages out noise.
  Quality: Omitting poor examples enables a better fit.
- **Manually enforcing a higher bias by using a less complex hypothesis space.**
- **Regularization**
  Automatic adjustment of the loss function to penalize model complexity.
Regularization

Motivation

Given the choice, can a learning algorithm favor a less-than-optimal hypothesis?
Regularization

Motivation

Given the choice, can a learning algorithm favor a less-than-optimal hypothesis?

Example: Linear Regression
Regularization

Motivation

Given the choice, can a learning algorithm favor a less-than-optimal hypothesis?

Example: Linear Regression
Regularization

Motivation

Given the choice, can a learning algorithm favor a less-than-optimal hypothesis?

Example: Linear Regression
Regularization

Motivation

Given the choice, can a learning algorithm favor a less-than-optimal hypothesis?

Example: Linear Regression
Regularization

Motivation

Given the choice, can a learning algorithm favor a less-than-optimal hypothesis?

Example: Linear Regression

Answer: No.
Regularization

Motivation

Change the optimal hypothesis $h^* \in H$ by augmenting the loss function $L$. 
Regularization
Motivation

Change the optimal hypothesis $h^* \in H$ by augmenting the loss function $L$.

Criteria regarding the desired $h^*$:
- Generalizability to unseen data.
- Simplicity (Occam’s razor)
- Stability wrt. to changes in $D$.
- Smoothness of the model function.

Desired mathematical properties:
- Differentiability
- Convexity
- Well-posedness of the inverse problem.
Regularization

Motivation

Change the optimal hypothesis $h^* \in H$ by augmenting the loss function $L$.

Criteria regarding the desired $h^*$:

- Generalizability to unseen data.
- Simplicity (Occam’s razor)
- Stability wrt. to changes in $D$.
- Smoothness of the model function.

Desired mathematical properties:

- Differentiability
- Convexity
- Well-posedness of the inverse problem.

Regularization is the theory of augmenting loss functions.
The term “regularization” derives from “regular”, a synonym for describing “smooth” model functions. [stackexchange]

The origins of regularization are found in inverse problem theory and solving ill-posed problems.

Regularization is only useful in settings where the set of examples $D$ is much smaller than the population of real-world objects $O$. From the examples, machine learning infers a hypothesis $h$ that is supposed to generalize to the entire population. Based on the assumption, that $D$ is a representative sample of the population, by choosing a simple, stable, and smooth hypothesis $h$, the risk of making errors on unseen objects is minimized.

In situations where $D$ covers (nearly) the entire population $O$, minimizing the loss $L$ takes precedence over simplicity, stability, and smoothness of the hypothesis $h$. One cannot expect the (natural) laws governing $O$ to be simple and smooth.
The scientific method to study physical systems:

→ Forward modeling. Proposition of a theory (discovery of a model). Suggestion of model parameters to predict observable system parameters.

← Inverse modeling. Use observable system parameters to identify the parameters of the model representing the system.

☐ Minimization of the set of model parameters that fully characterize the system.
The scientific method to study physical systems:

→ **Forward modeling.** Proposition of a theory (discovery of a model). Suggestion of model parameters to predict observable system parameters.

← **Inverse modeling.** Use observable system parameters to identify the parameters of the model representing the system.

□ **Minimization.** Minimization of the set of model parameters that fully characterize the system.
The scientific method to study physical systems:

- **Forward modeling.** Proposition of a theory (discovery of a model). Suggestion of model parameters to predict observable system parameters.

- **Inverse modeling.** Use observable system parameters to identify the parameters of the model representing the system.

- Minimization of the set of model parameters that fully characterize the system.
Regularization
Machine Learning as an Inverse Problem

The scientific method to study physical systems:

→ Forward modeling. Proposition of a theory (discovery of a model). Suggestion of model parameters to predict observable system parameters.

← Inverse modeling. Use observable system parameters to identify the parameters of the model representing the system.

- Minimization of the set of model parameters that fully characterize the system.
Regularization
Machine Learning as an Inverse Problem

Historical inverse problems: [Google scholar]:

- Modeling the motions of the planets from observations of the night sky.
  Aristotle → Copernicus → Kepler → Newton → Einstein [NASA 2009]

- Modeling the inner composition of the Earth from observations at its surface.

- Modeling and predicting the weather based on past observations.

Retrograde motion of Mars within 35 weeks.
Regularization
Machine Learning as an Inverse Problem

Historical inverse problems [Google scholar] :

- Modeling the motions of the planets from observations of the night sky. Aristotle → Copernicus → Kepler → Newton → Einstein [NASA 2009]
- Modeling the inner composition of the Earth from observations at its surface.
- Modeling and predicting the weather based on past observations.

Practical considerations:

- Observable parameters may be insufficient to explain the system’s behavior.
- Measurements contain errors dependent on the accuracy of the instrument.
- Inverse problems are often ill-posed due to discretization.
Regularization
Well-Posed vs. Ill-Posed Problems

**Definition 10 (Well-posed Problem)** [Hadamard 1902])

A mathematical problem is called well-posed if

1. a solution exists,
2. the solution is unique,
3. the solution’s behavior changes continuously with the initial conditions.

Otherwise, the problem is called ill-posed.
Regularization
Well-Posed vs. Ill-Posed Problems

Definition 10 (Well-posed Problem [Hadamard 1902])
A mathematical problem is called well-posed if

1. a solution exists,
2. the solution is unique,
3. the solution’s behavior changes continuously with the initial conditions.

Otherwise, the problem is called ill-posed.

Linear regression:

- Existence and uniqueness of a solution depends on the rank of $X$.

→ Use the pseudoinverse $(X^T X)^{-1} X^T$.

- The solution does not depend continuously on $X$.

- Introducing noise or removing data may strongly affect the solution.

→ A possible way to find a solution is regularization. [Tikhonov 1977]
Remarks:

- According to Hadamard’s philosophy, ill-posed problems are actually ill-posed, in the sense that the underlying model is wrong.

- Hadamard thought that ill-posed problems are a pure mathematical phenomenon and that all real-life problems are well-posed. However, in the second half of the century a number of very important real-life problems were found to be ill-posed. In particular, ill-posed problems arise when one tries to reverse the cause-effect relations: to find unknown causes from known consequences. Even if the cause-effect relationship forms a one-to-one mapping, the problem of inverting it can be ill-posed. [Vapnik 2000]

- Supervised learning has been shown to be an inverse problem. [de Vito 2005]
Regularization

Definition

Let $L(w)$ denote a loss function used to optimize the parameters $w$ of a model function $y(x)$. Regularization introduces a trade-off between model complexity and inductive bias:

$$\mathcal{L}(w) = L(w) + \lambda \cdot R(w),$$

where $\lambda \geq 0$ controls the impact of the regularization term $R(w) \geq 0$. $\mathcal{L}$ is called “objective function”.

Regularization

Definition

Let $L(w)$ denote a loss function used to optimize the parameters $w$ of a model function $y(x)$. Regularization introduces a trade-off between model complexity and inductive bias:

$$L(w) = L(w) + \lambda \cdot R(w),$$

where $\lambda \geq 0$ controls the impact of the regularization term $R(w) \geq 0$. $L$ is called “objective function”.

Example:

- $y(x) = w_0 + \sum_{i=1}^{6} w_i \cdot x^i$; $L(w) = \text{RSS}(w)$
- $\lambda = 0$
- $R(w) = 0$

$\Rightarrow \hat{w} = (-0.7, 14.1, -63.4, 114.8, -39.8, -84.1, 59.9)^T$
Regularization
Definition

Let $L(w)$ denote a loss function used to optimize the parameters $w$ of a model function $y(x)$. Regularization introduces a trade-off between model complexity and inductive bias:

$$L(w) = L(w) + \lambda \cdot R(w),$$

where $\lambda \geq 0$ controls the impact of the regularization term $R(w) \geq 0$. $L$ is called “objective function”.

Example:

- $y(x) = w_0 + \sum_{i=1}^{6} w_i \cdot x^i$; $L(w) = \text{RSS}(w)$
- $\lambda = 1$
- $R(w) = 0 \cdot |w_1| + 100 \cdot |w_2| + \ldots + 100 \cdot |w_6|$

$\Rightarrow \hat{w} = (-0.2, 0.6, 0, 0, 0, 0, 0)^T$
Regularization

Definition

Let $L(w)$ denote a loss function used to optimize the parameters $w$ of a model function $y(x)$. Regularization introduces a trade-off between model complexity and inductive bias:

$$\mathcal{L}(w) = L(w) + \lambda \cdot R(w),$$

where $\lambda \geq 0$ controls the impact of the regularization term $R(w) \geq 0$. $\mathcal{L}$ is called “objective function”.

Example:

- $y(x) = w_0 + \sum_{i=1}^{6} w_i \cdot x^i$; \( L(w) = \text{RSS}(w) \)
- $\lambda = 1/1000$
- $R(w) = 0 \cdot |w_1| + 100 \cdot |w_2| + \ldots + 100 \cdot |w_6|$
- $\hat{w} = (-0.1, 0.6, 2.1, -1.0, -3.1, -0.5, 2.5)^T$
Regularization

Definition

Let $L(w)$ denote a loss function used to optimize the parameters $w$ of a model function $y(x)$. Regularization introduces a trade-off between model complexity and inductive bias:

$$\mathcal{L}(w) = L(w) + \lambda \cdot R(w),$$

where $\lambda \geq 0$ controls the impact of the regularization term $R(w) \geq 0$. $\mathcal{L}$ is called “objective function”.

Observations:

- Model complexity (partially) depends on the magnitude of the weights $w$.
- Minimizing $L(w)$ places no bounds on the weights $w$.
- Regularization introduces a “counterweight” $\lambda \cdot R(w)$ that grows with $w$.
- Except $\lambda$, no additional hyperparameters are introduced.
Regularization

Definition

Let $L(w)$ denote a loss function used to optimize the parameters $w$ of a model function $y(x)$. Regularization introduces a trade-off between model complexity and inductive bias:

$$
\mathcal{L}(w) = L(w) + \lambda \cdot R(w),
$$

where $\lambda \geq 0$ controls the impact of the regularization term $R(w) \geq 0$. $\mathcal{L}$ is called “objective function”.

Regularization using the vector norm:

- **Lasso regression.**
  $$
  R_{||\vec{w}||_1}(w) = \sum_{i=1}^{p} |w_i|
  $$

- **Ridge regression.**
  $$
  R_{||\vec{w}||_2^2}(w) = \sum_{i=1}^{p} w_i^2 = \vec{w}^T \vec{w}
  $$
Remarks:

- “Lasso” is an acronym for “least absolute shrinkage and selection operator” and “ridge” a metaphor describing planes shaped like a hill range the highest point of which is not easily discernible.

- Ridge regression predates lasso regression. It is also known as weight decay in machine learning, and with multiple independent discoveries, it is variously known as the Tikhonov-Miller method, the Phillips-Twomey method, the constrained linear inversion method, and the method of linear regularization. [Wikipedia]

- \( \| \cdot \|_k \) denotes the vector norm operator:

\[
\| \mathbf{x} \|_k \equiv \left( \sum_{i=1}^{p} |x_i|^k \right)^{1/k},
\]

where \( k \in [1, \infty) \) and \( p \) is the dimensionality of vector \( \mathbf{x} \).

By convention, \( \| \cdot \| \) (omitting the subscript) typically refers to the Euclidean norm \( (k = 2) \).

- The regularization term constrains the magnitude of the direction vector of the hyperplane, progressively reducing the hyperplane’s steepness as \( \lambda \) increases. The intercept \( w_0 \) is adjusted accordingly through minimization of \( \mathcal{L}(\mathbf{w}) \) but must not be part of the regularization term itself, which would lead to an incorrect solution.

To denote the difference, we write \( \mathbf{w} \equiv (w_0, w_1, \ldots, w_p)^T \) to refer to the entire parameter vector (the actual hypothesis), and \( \mathbf{\tilde{w}} \equiv (w_1, \ldots, w_p)^T \) for the direction vector excluding \( w_0 \).
Regularization

Illustration

\[ \mathcal{L}(w) = L(w) + \lambda \cdot R(w) \]

Example:
Regularization
Illustration

\[ \mathcal{L}(w) = L(w) + \lambda \cdot R(w) \]

Example:
\[ \mathcal{L}(w) = L(w) + \lambda \cdot R(w) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(w) = L(w) + \lambda \cdot R(w) \]

Example:
Regularization
Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R(\mathbf{w}) \]

Example:
Regularization
Illustration: Lasso Regression

\[ R_{||\tilde{w}||_1}(w) = \sum_{i=1}^{p} |w_i| \]

Example:
Regularization
Illustration: Lasso Regression

\[ R_{||\vec{w}||_1}(\vec{w}) = \sum_{i=1}^{p} |w_i| \]

Example:
Regularization
Illustration: Lasso Regression

\[ R_{||\vec{w}||_1}(\vec{w}) = \sum_{i=1}^{p} |w_i| \]

Example:
Regularization

Illustration: Lasso Regression

\[ R_{\|\mathbf{w}\|_1}(\mathbf{w}) = \sum_{i=1}^{p} |w_i| \]

Example:
Regularization
Illustration: Ridge Regression

\[
R_{||\vec{w}||_2^2}(\vec{w}) = \sum_{i=1}^{p} w_i^2 = \vec{w}^T \vec{w}
\]

Example:
\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(w) = L(w) + \lambda \cdot R_{||\vec{w}||^2}(w) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{||\mathbf{w}||_2^2}(\mathbf{w}) \]

Example:
Regularization
Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{||\mathbf{w}||_1}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(w) = L(w) + \lambda \cdot R_{||\bar{w}||_1}(w) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{\|\mathbf{w}\|_1}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{||\mathbf{w}||^2}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{||\bar{\mathbf{w}}||_2^2}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{\|\mathbf{w}\|_2^2}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{||\mathbf{w}||^2}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(w) = L(w) + \lambda \cdot R_{||w||^2}(w) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{||\mathbf{w}||^2}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{\|\mathbf{w}\|_2^2}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(\mathbf{w}) = L(\mathbf{w}) + \lambda \cdot R_{\|\mathbf{w}\|_2^2}(\mathbf{w}) \]

Example:
Regularization

Illustration

\[ \mathcal{L}(w) = L(w) + \lambda \cdot R_{\|\bar{w}\|_2^2}(w) \]

Example:
Remarks:

- The exemplified contour line plots of the parameter space (hypothesis space) show two-dimensional projections of the three-dimensional convex loss function (e.g., RSS) for a given set of example data, as well as the two regularization functions lasso $R_{||w||_1}$ and ridge $R_{||w||_2^2}$, respectively, whose shapes do not depend on the data. A contour line is a curve along which the respective function has a constant value.

- The exemplified loss function is minimal at the cross. Without regularization (e.g., $\lambda = 0$), the weights associated with the minimum would be the result of a linear regression. By adding the regularization term $\lambda \cdot R(w)$, with $\lambda > 0$, the joint minimum of the two functions is found closer to the origin of the parameter space than the minimum of the loss function.

- The choice of $\lambda$ determines how much closer the joint minimum is to the origin of the parameter space; the higher, the closer, and thus the smaller the parameters $w$.

- For a given $\lambda$, the new minimum is found where a contour line of the loss function tangents that of the regularization function.

- The choice of the regularization function (e.g., lasso $R_{||w||_1}$ or ridge $R_{||w||_2^2}$) determines the trajectory the minimum takes towards the origin as a function of $\lambda$. [stackexchange]
Remarks: (continued)

- A key difference between lasso and ridge regression is that, with lasso regression, parameters can be reduced to zero, eliminating the corresponding feature from the model function. With ridge regression, a parameter will be reduced to zero if, and only if, the minimum of the loss function is found on that parameter’s axis.

- Lasso regression divides the parameter space into regions; when the loss function’s minimum is found in the ones shaded gray, with increasing $\lambda$, the parameter value of the closest axis is eventually reduced to zero.

- Using lasso regression renders the associated inverse problem ill-posed, since it’s solutions may not be unique. Moreover, the optimal parameters cannot be computed via a direct method.
Regularization

Regularized Linear Regression

- Given $x$, predict a real-valued output under a linear model function:
  \[ y(x) = w_0 + \sum_{j=1}^{p} w_j \cdot x_j \]
- Vector notation with $x_0 = 1$ and $w = (w_0, w_1, \ldots, w_p)^T$:
  \[ y(x) = w^T x \]
- Given $x_1, \ldots, x_n$, assess goodness of fit of the objective function:
  \[
  \mathcal{L}(w) = \text{RSS}(w) + \lambda \cdot R_{||w||^2}(w) = \sum_{i=1}^{n} (y_i - w^T x_i)^2 + \lambda \cdot \bar{w}^T \bar{w}
  \] (1)
- Estimate $w$ by minimizing the residual sum of squares:
  \[
  \hat{w} = \arg\min_{w \in \mathbb{R}^{p+1}} \mathcal{L}(w)
  \] (2)
Regularization
Regularized Linear Regression

- Let $X$ denote the $n \times (p + 1)$ matrix, where row $i$ is $(1 \ x_i^T)$, $x_i \in D$.

  Let $y$ denote the $n$-vector of outputs in the training set $D$.

  \[ \mathcal{L}(w) = (y - Xw)^T(y - Xw) + \lambda \cdot \vec{w}^T \vec{w} \]
Regularization
Regularized Linear Regression

- Let $X$ denote the $n \times (p + 1)$ matrix, where row $i$ is $(1 \ x_i^T)$, $x_i \in D$.

- Let $y$ denote the $n$-vector of outputs in the training set $D$.

\[ L(w) = (y - Xw)^T(y - Xw) + \lambda \cdot \vec{w}^T \vec{w} \]

- Minimizing $L(w)$ by a direct method:

\[
\frac{\partial L(w)}{\partial w} = -2X^T(y - Xw) + 2\lambda \cdot (0_{\vec{w}}) = 0
\]

\[
X^T(y - Xw) - \lambda \cdot (0_{\vec{w}}) = 0
\]

\[
\Leftrightarrow (X^T X + \lambda \cdot \text{diag}(0, 1, \ldots, 1))w = X^T y
\]

\[
\Leftrightarrow w = (X^T X + \text{diag}(0, \lambda, \ldots, \lambda))^{-1} X^T y
\]
Regularization

Regularized Linear Regression

- Let $X$ denote the $n \times (p + 1)$ matrix, where row $i$ is $(1 \ x_i^T)$, $x_i \in D$.

  Let $y$ denote the $n$-vector of outputs in the training set $D$.

  $$\mathcal{L}(w) = (y - Xw)^T(y - Xw) + \lambda \cdot \vec{w}^T \vec{w}$$

- Minimizing $\mathcal{L}(w)$ by a direct method:

  $$\frac{\partial \mathcal{L}(w)}{\partial w} = -2X^T(y - Xw) + 2\lambda \cdot (\vec{0}) = 0$$

  $$X^T(y - Xw) - \lambda \cdot (\vec{0}) = 0$$

  $$\Leftrightarrow (X^TX + \lambda \cdot \text{diag}(0, 1, \ldots, 1))w = X^Ty$$

  $$\Leftrightarrow w = (X^TX + \text{diag}(0, \lambda, \ldots, \lambda))^{-1}X^Ty$$

  Normal eqns.

  Conditioning the moment matrix $X^TX$ [Wikipedia 1, 2, 3]

If $\lambda > 0$. 

---

ML:III-128 Linear Models © POTTHAST/STEIN 2021
Regularization
Regularized Linear Regression

- Let $X$ denote the $n \times (p+1)$ matrix, where row $i$ is $(1 \ x_i^T)$, $x_i \in D$.

  Let $y$ denote the $n$-vector of outputs in the training set $D$.

  \[ L(w) = (y - Xw)^T(y - Xw) + \lambda \cdot \vec{w}^T \vec{w} \]

- Minimizing $L(w)$ by a direct method:

  \[
  \frac{\partial L(w)}{\partial w} = -2X^T(y - Xw) + 2\lambda \cdot \left( \frac{0}{\vec{w}} \right) = 0
  \]

  \[ X^T(y - Xw) - \lambda \cdot \left( \frac{0}{\vec{w}} \right) = 0 \]

  \[ \Leftrightarrow \ (X^T X + \lambda \cdot \text{diag}(0, 1, \ldots, 1))w = X^T y \]

  \[ \Leftrightarrow \ \hat{w} \equiv w = (X^T X + \text{diag}(0, \lambda, \ldots, \lambda))^{-1} X^T y \quad \text{Normal eqns.} \]

  If $\lambda > 0$. Conditioning the moment matrix $X^T X$ [Wikipedia 1, 2, 3]

  \[
  \hat{y}(x_i) = \hat{w}^T x_i \quad \text{Regression function with least squares estimator} \ \hat{w}.
  \]
Regularization

Regularized Linear Regression: Hyperparameter $\lambda$

The regularization parameter $\lambda$ penalizes high weights $w_i$:

$$y(x) = w_0 + \sum_{i=1}^{p} w_i \cdot x^i = \mathbf{w}^T \mathbf{x}$$

$$\mathcal{L}(\mathbf{w}) = \sum_{i=1}^{n} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 + \lambda \cdot \mathbf{w}^T \mathbf{w}$$
Regularization

Regularized Linear Regression: Hyperparameter $\lambda$

\[ \lambda = 0 \quad \lambda = 0.2 \quad \lambda = 0.7 \]

“No black-box procedures for choosing the regularization parameter $\lambda$ are available, and most likely will never exist.” [Hansen and Hanke 1993]

Heuristics:

- Model selection for a choice of $\lambda_1, \ldots, \lambda_m$. [stackoverflow]
- More advanced heuristics have been proposed. [Bauer and Lukas 2011]