The double descent phenomenon

Generalization of overparametrized models

Geelon So, agso@eng.ucsd.edu
DSC291 Machine Learning — November 8, 2022

Review: classical statistical learning framework

A **learner** would like to answer a question about the world.

Review: classical statistical learning framework

A **learner** would like to answer a question about the world.

1. The learner selects a **model**—a family of possible explanations/hypotheses.

Review: classical statistical learning framework

A **learner** would like to answer a question about the world.

1. The learner selects a **model**—a family of possible explanations/hypotheses.
2. The learner collects **data** from the world.

Review: classical statistical learning framework

A **learner** would like to answer a question about the world.

1. The learner selects a **model**—a family of possible explanations/hypotheses.
2. The learner collects **data** from the world.
3. The learner then **fits** the model to the data.

Review: classical statistical learning framework

A **learner** would like to answer a question about the world.

1. The learner selects a **model**—a family of possible explanations/hypotheses.
2. The learner collects **data** from the world.
3. The learner then **fits** the model to the data.

The model's ability to **generalize** is how well it accounts for out-of-sample data.

Learning through risk minimization

The standard approach to learning

Learning through risk minimization

The standard approach to learning

1. Select a **model** $\mathcal{H}$.

Learning through risk minimization

The standard approach to learning

1. Select a **model** $\mathcal{H}$.
2. Define the **risk** of a hypothesis $h \in \mathcal{H}$ as:

$$R(h) = \text{a measure of how poorly } h \text{ explains the world.}$$

Learning through risk minimization

The standard approach to learning

1. Select a **model** $\mathcal{H}$.
2. Define the **risk** of a hypothesis $h \in \mathcal{H}$ as:

$$R(h) = \text{a measure of how poorly } h \text{ explains the world.}$$

- The goal is to find the *best-in-class explanation* $h^* = \arg\min_{h \in \mathcal{H}} R(h)$.

Learning through risk minimization

The standard approach to learning

1. Select a **model** $\mathcal{H}$.
2. Define the **risk** of a hypothesis $h \in \mathcal{H}$ as:

$$R(h) = \text{a measure of how poorly } h \text{ explains the world.}$$

- The goal is to find the *best-in-class explanation* $h^* = \arg\min_{h \in \mathcal{H}} R(h)$.
- The *model bias* measures the risk $R(h^*)$ of the best explanation.

Learning through risk minimization

The standard approach to learning

1. Select a **model** $\mathcal{H}$.
2. Define the **risk** of a hypothesis $h \in \mathcal{H}$ as:

$$R(h) = \text{a measure of how poorly } h \text{ explains the world.}$$

- The goal is to find the *best-in-class explanation* $h^* = \arg\min_{h \in \mathcal{H}} R(h)$.
- The *model bias* measures the risk $R(h^*)$ of the best explanation.
- We generally cannot compute the risk directly, but we can estimate it.

Learning through risk minimization

The standard approach to learning

1. Select a **model** $\mathcal{H}$.
2. Define the **risk** of a hypothesis $h \in \mathcal{H}$ as:

$$R(h) = \text{a measure of how poorly } h \text{ explains the world.}$$

   - The goal is to find the *best-in-class explanation* $h^* = \arg\min_{h \in \mathcal{H}} R(h)$.
   - The *model bias* measures the risk $R(h^*)$ of the best explanation.
   - We generally cannot compute the risk directly, but we can estimate it.
3. Construct a **risk estimation** procedure that finds an estimate $\hat{R}(h)$ of $R(h)$.

Learning through risk minimization

The standard approach to learning

1. Select a **model** $\mathcal{H}$.
2. Define the **risk** of a hypothesis $h \in \mathcal{H}$ as:

$$R(h) = \text{a measure of how poorly } h \text{ explains the world.}$$

   - The goal is to find the *best-in-class explanation* $h^* = \arg\min_{h \in \mathcal{H}} R(h)$.
   - The *model bias* measures the risk $R(h^*)$ of the best explanation.
   - We generally cannot compute the risk directly, but we can estimate it.

3. Construct a **risk estimation** procedure that finds an estimate $\hat{R}(h)$ of $R(h)$.
4. **Minimize** the risk estimator:

$$\hat{h} := \arg\min_{h \in \mathcal{H}} \hat{R}(h).$$

Generalization theory through empirical risk minimization

We can decompose the risk $R(h)$ as:

$$R(h) = \underbrace{\hat{R}(h) - \hat{R}(h^*)}_{\text{estimated gap}} + \underbrace{\left(R(h) - \hat{R}(h)\right)}_{\text{estimation error for } h} + \underbrace{\left(\hat{R}(h^*) - R(h^*)\right)}_{\text{estimation error for } h^*} + \underbrace{R(h^*)}_{\text{model bias}}$$

Generalization theory through empirical risk minimization

We can decompose the risk $R(h)$ as:

$$R(h) = \underbrace{\hat{R}(h) - \hat{R}(h^*)}_{\text{estimated gap}} + \underbrace{\left(R(h) - \hat{R}(h)\right)}_{\text{estimation error for } h} + \underbrace{\left(\hat{R}(h^*) - R(h^*)\right)}_{\text{estimation error for } h^*} + \underbrace{R(h^*)}_{\text{model bias}}$$

- For the empirical risk minimizer $\hat{h}$, the *estimated gap* term is non-positive, so:

$$R(\hat{h}) \leq \text{estimation error terms} + \text{model bias term}.$$

Results from classical generalization theory

Generalization theory tends to give us bounds on the estimation error term so that:

$$R(\hat{h}) \leq \sqrt{\frac{\text{capacity of model}}{\text{amount of training data}}} + \text{model bias}.$$

Results from classical generalization theory

Generalization theory tends to give us bounds on the estimation error term so that:

$$R(\hat{h}) \leq \sqrt{\frac{\text{capacity of model}}{\text{amount of training data}}} + \text{model bias.}$$

- The **capacity** of $\mathcal{H}$ measures how many worlds $\mathcal{H}$ can explain.

Results from classical generalization theory

Generalization theory tends to give us bounds on the estimation error term so that:

$$R(\hat{h}) \leq \sqrt{\frac{\text{capacity of model}}{\text{amount of training data}}} + \text{model bias.}$$

- The **capacity** of $\mathcal{H}$ measures how many worlds $\mathcal{H}$ can explain.
- Generally, the bias of a model increases as its capacity shrinks.

Results from classical generalization theory

Generalization theory tends to give us bounds on the estimation error term so that:

$$R(\hat{h}) \leq \sqrt{\frac{\text{capacity of model}}{\text{amount of training data}}} + \text{model bias.}$$

- The **capacity** of $\mathcal{H}$ measures how many worlds $\mathcal{H}$ can explain.
- Generally, the bias of a model increases as its capacity shrinks.
  - This leads to the **bias-variance tradeoff**.

Bias-variance tradeoff

Intuition: how the bias of $\mathcal{H}$ relate to the capacity of $\mathcal{H}$.

Bias-variance tradeoff

Intuition: how the bias of $\mathcal{H}$ relate to the capacity of $\mathcal{H}$.

- **Small capacity:** if $\mathcal{H}$ cannot explain many worlds, it may poorly explain the one in which the learner lives. This leads to a large bias term.

Bias-variance tradeoff

Intuition: how the bias of $\mathcal{H}$ relate to the capacity of $\mathcal{H}$.

- **Small capacity:** if $\mathcal{H}$ cannot explain many worlds, it may poorly explain the one in which the learner lives. This leads to a large bias term.
- **Large capacity:** if many (very different) explanations account for what the learner sees, how to pick among these explanations? This leads to a large variance term.

Classical bias-variance tradeoff

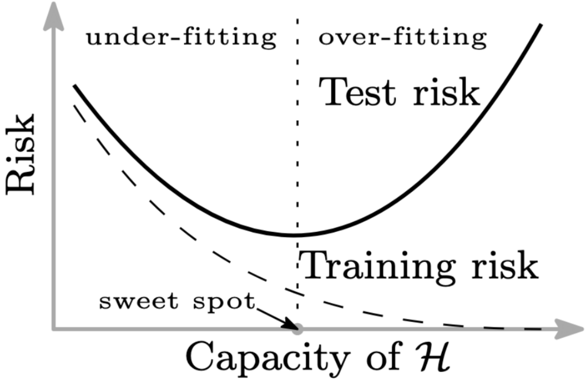


Figure 1: Belkin et al. (2018)

Model selection problem

Question. How complicated of a model should you try to fit?

Model selection problem

Question. How complicated of a model should you try to fit?

- Classical statistics says not too large: try to find the 'sweet spot'.

Model selection problem

Question. How complicated of a model should you try to fit?

- Classical statistics says not too large: try to find the 'sweet spot'.
- However, in modern machine learning, we often fit very over-parameterized models and achieve good generalization.

Model selection problem

Question. How complicated of a model should you try to fit?

- Classical statistics says not too large: try to find the 'sweet spot'.
- However, in modern machine learning, we often fit very over-parameterized models and achieve good generalization.
  - The capacity of neural nets often allow for training loss to be driven down to zero (that is, the model *interpolates* the training data).

Double descent phenomenon

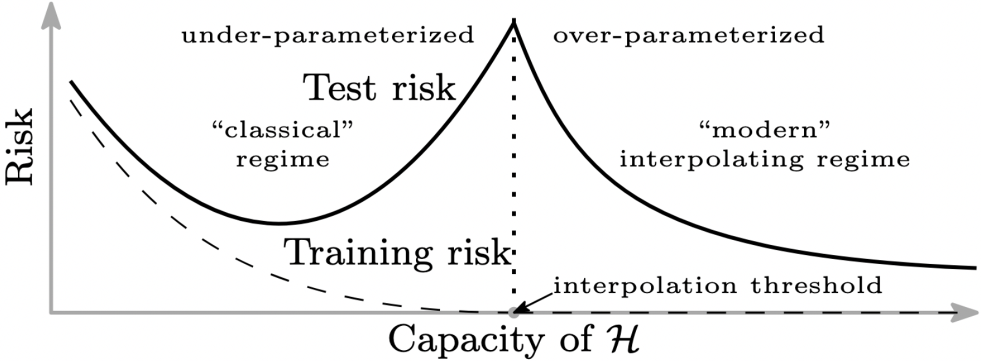


Figure 2: In the 'modern' interpolating regime, increasing model capacity often empirically leads to better generalization, Belkin et al. (2018).

Double descent a robust phenomenon

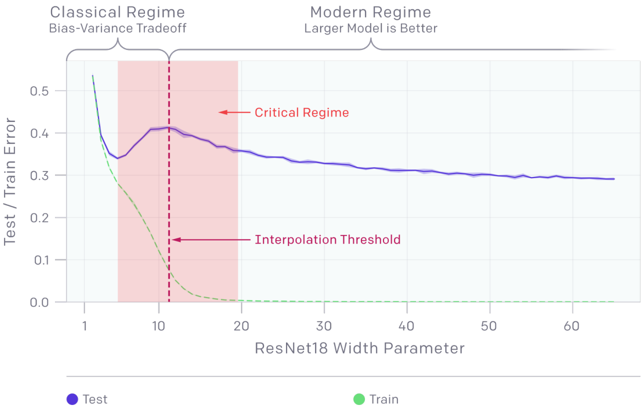


Figure 3: Double descent is observed across many models, tasks, optimizers, training time, and noise levels. Pictured is the train/test error for family of ResNet18 on CIFAR-10 (Nakkiran et al., 2021).

Generalization theory: what's missing?

- In an over-parameterized model where many explanations equally account for the training data, how does the learner select one?
  - Generalization also depends on how we regularize and optimize.

Algorithms without double descent?

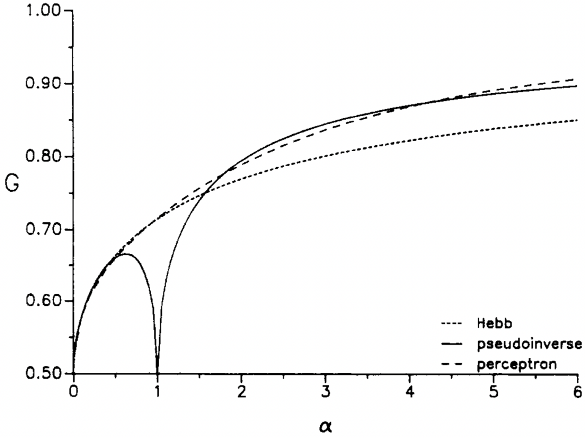


Figure 4: Generalization curves for different methods of learning a classifier on Boolean data $\{-1, +1\}^N$ using a dataset of size αN (Opper et al., 1990).

Double descent in ordinary least squares: warm-up

One explanation of double descent

Belkin et al. (2020) examines double descent through the lens of signal-to-noise ratio.

One explanation of double descent

Belkin et al. (2020) examines double descent through the lens of signal-to-noise ratio.

Intuition for ordinary least squares (OLS): if the number of data points is around the number of dimensions, then likely there are directions with a low signal-to-noise ratio. OLS will overfit those directions to noise.

One explanation of double descent

Belkin et al. (2020) examines double descent through the lens of signal-to-noise ratio.

Intuition for ordinary least squares (OLS): if the number of data points is around the number of dimensions, then likely there are directions with a low signal-to-noise ratio. OLS will overfit those directions to noise.

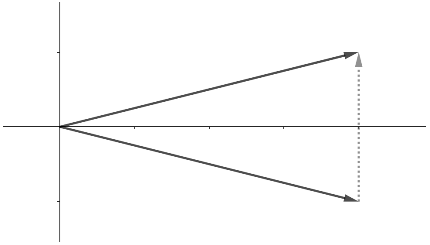


Figure 5: The instances x_1 and x_2 (black) provides good signal along the horizontal direction, but poor signal along the vertical direction.

Linear regression problem

Problem. Suppose nature generates data as follows:

$$y = x^\top \beta + \varepsilon,$$

Linear regression problem

Problem. Suppose nature generates data as follows:

$$y = x^\top \beta + \varepsilon,$$

- the covariates $x \in \mathbb{R}^d$ are d dimensional

Linear regression problem

Problem. Suppose nature generates data as follows:

$$y = x^\top \beta + \varepsilon,$$

- the covariates $x \in \mathbb{R}^d$ are d dimensional
- the noise $\varepsilon \in \mathbb{R}$ is drawn from $\mathcal{N}(0, \sigma^2)$

Linear regression problem

Problem. Suppose nature generates data as follows:

$$y = x^\top \beta + \varepsilon,$$

- the covariates $x \in \mathbb{R}^d$ are d dimensional
- the noise $\varepsilon \in \mathbb{R}$ is drawn from $\mathcal{N}(0, \sigma^2)$
- there is a true regressor β, but it is unknown to the learner

Linear regression problem

Problem. Suppose nature generates data as follows:

$$y = x^\top \beta + \varepsilon,$$

- the covariates $x \in \mathbb{R}^d$ are d dimensional
- the noise $\varepsilon \in \mathbb{R}$ is drawn from $\mathcal{N}(0, \sigma^2)$
- there is a true regressor β, but it is unknown to the learner

Goal. The goal of the learner is to use data to give an estimate $\hat{\beta}$ of β minimizing:

$$\|\hat{\beta} - \beta\|^2.$$

Linear regression in the interpolating regime

When the number of parameters d is at least the number of data points n, we can always perfectly fit a linear regressor:

$$\mathbf{Y} = \mathbf{X}\hat{\beta},$$

where $\mathbf{X} \in \mathbb{R}^{n \times d}$, $\mathbf{Y} \in \mathbb{R}^{n}$, and $\hat{\beta} \in \mathbb{R}^{d}$.

Linear regression in the interpolating regime

When the number of parameters d is at least the number of data points n, we can always perfectly fit a linear regressor:

$$\mathbf{Y} = \mathbf{X}\hat{\beta},$$

where $\mathbf{X} \in \mathbb{R}^{n \times d}$, $\mathbf{Y} \in \mathbb{R}^n$, and $\hat{\beta} \in \mathbb{R}^d$.

- In fact, if $d > n$, then there are infinitely many interpolating $\hat{\beta}$'s.

Linear regression in the interpolating regime

When the number of parameters d is at least the number of data points n, we can always perfectly fit a linear regressor:

$$\mathbf{Y} = \mathbf{X}\hat{\beta},$$

where $\mathbf{X} \in \mathbb{R}^{n \times d}$, $\mathbf{Y} \in \mathbb{R}^n$, and $\hat{\beta} \in \mathbb{R}^d$.

- In fact, if $d > n$, then there are infinitely many interpolating $\hat{\beta}$'s.
- If we fit $\hat{\beta}$ using OLS, we obtain a specific choice:

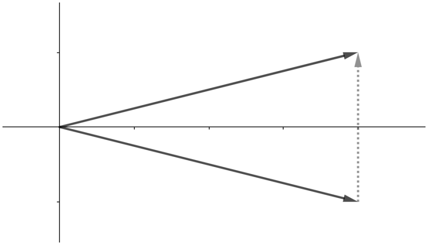
$$\hat{\beta} = \mathbf{X}^{+}\mathbf{Y},$$

where $\mathbf{X}^{+}$ is the Moore-Penrose pseudoinverse of $\mathbf{X}$. In this setting, $\mathbf{X}\mathbf{X}^{+} = \mathbf{I}_d$.

Linear regression in the interpolating regime

Question. How poorly can $\hat{\beta}$ estimate β in the interpolating regime?

A 2D example

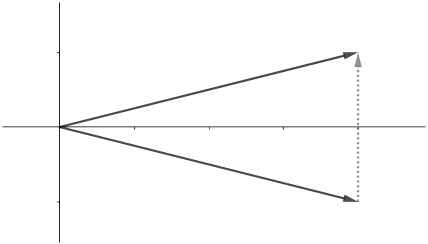


Setting. Consider data in $\mathbb{R}^2$ generated by:

$$y = x^\top e_1 + \varepsilon,$$

where e_1 is the first basis direction and ε is Gaussian.

A 2D example



Setting. Consider data in $\mathbb{R}^2$ generated by:

$$y = x^\top e_1 + \varepsilon,$$

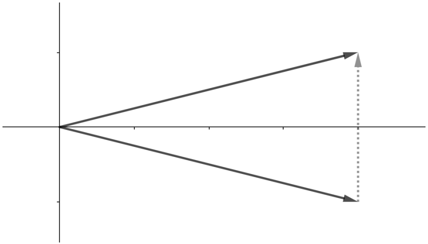
where e_1 is the first basis direction and ε is Gaussian.

Data. Suppose we are given two data points:

$$x_A = \begin{bmatrix} 1 \\ -\delta \end{bmatrix} \qquad x_B = \begin{bmatrix} 1 \\ +\delta \end{bmatrix}$$

$$y_A = 1 + \varepsilon_A \qquad y_B = 1 + \varepsilon_B$$

A 2D example



Setting. Consider data in $\mathbb{R}^2$ generated by:

$$y = x^\top e_1 + \varepsilon,$$

where e_1 is the first basis direction and ε is Gaussian.

Data. Suppose we are given two data points:

$$x_A = \begin{bmatrix} 1 \\ -\delta \end{bmatrix} \qquad x_B = \begin{bmatrix} 1 \\ +\delta \end{bmatrix}$$

$$y_A = 1 + \varepsilon_A \qquad y_B = 1 + \varepsilon_B$$

Solution. OLS estimate $\hat{\beta}$ is:

$$\hat{\beta}_1 = 1 + \frac{\varepsilon_A + \varepsilon_B}{2} \qquad \hat{\beta}_2 = \frac{1}{\delta}\frac{\varepsilon_B - \varepsilon_A}{2}.$$

Computation of 2D example

Here, $\beta = e_1$ and OLS estimate $\hat{\beta}$ is:

$$\hat{\beta}_1 = 1 + \frac{\varepsilon_A + \varepsilon_B}{2} \qquad \hat{\beta}_2 = \frac{1}{\delta}\frac{\varepsilon_B - \varepsilon_A}{2}.$$

- Notice that: $e_1 = \dfrac{x_A + x_B}{2}$ and $\delta e_2 = \dfrac{x_B - x_A}{2}$.

Computation of 2D example

Here, $\beta = e_1$ and OLS estimate $\hat{\beta}$ is:

$$\hat{\beta}_1 = 1 + \frac{\varepsilon_A + \varepsilon_B}{2} \qquad \hat{\beta}_2 = \frac{1}{\delta}\frac{\varepsilon_B - \varepsilon_A}{2}.$$

- Notice that: $e_1 = \frac{x_A + x_B}{2}$ and $\delta e_2 = \frac{x_B - x_A}{2}$.
- The vector e_1 has to explain β_1 and some noise:

$$\frac{y_A + y_B}{2} = 1 + \frac{\varepsilon_A + \varepsilon_B}{2}.$$

Computation of 2D example

Here, $\beta = e_1$ and OLS estimate $\hat{\beta}$ is:

$$\hat{\beta}_1 = 1 + \frac{\varepsilon_A + \varepsilon_B}{2} \qquad \hat{\beta}_2 = \frac{1}{\delta}\frac{\varepsilon_B - \varepsilon_A}{2}.$$

- Notice that: $e_1 = \dfrac{x_A + x_B}{2}$ and $\delta e_2 = \dfrac{x_B - x_A}{2}$.
- The vector e_1 has to explain β_1 and some noise:

$$\frac{y_A + y_B}{2} = 1 + \frac{\varepsilon_A + \varepsilon_B}{2}.$$

- But, the small vector δe_2 also has to explain a (relatively large) part of the noise:

$$\frac{y_B - y_A}{2} = \frac{\varepsilon_B - \varepsilon_A}{2}.$$

Double descent from 2D example

Suppose instead that x_A and x_B were actually $(d+1)$-dimensional vectors:

$$
\begin{aligned}
x_A &= \begin{bmatrix} 1 & -\delta/\sqrt{d} & \cdots & -\delta/\sqrt{d} \end{bmatrix}^\top \\
x_B &= \begin{bmatrix} 1 & +\delta/\sqrt{d} & \cdots & +\delta/\sqrt{d} \end{bmatrix}^\top .
\end{aligned}
$$

Double descent from 2D example

Suppose instead that x_A and x_B were actually $(d+1)$-dimensional vectors:

$$x_A = \begin{bmatrix} 1 & -\delta/\sqrt{d} & \cdots & -\delta/\sqrt{d} \end{bmatrix}^\top$$
$$x_B = \begin{bmatrix} 1 & +\delta/\sqrt{d} & \cdots & +\delta/\sqrt{d} \end{bmatrix}^\top .$$

- The same part of the noise $\frac{\varepsilon_B - \varepsilon_A}{2}$ needs to be explained by a vector:

$$\frac{x_B - x_A}{2} = \frac{\delta}{\sqrt{d}} \cdot \begin{bmatrix} 0 & 1 & \cdots & 1 \end{bmatrix}^\top ,$$

Double descent from 2D example

Suppose instead that x_A and x_B were actually $(d+1)$-dimensional vectors:

$$x_A = \begin{bmatrix} 1 & -\delta/\sqrt{d} & \cdots & -\delta/\sqrt{d} \end{bmatrix}^\top$$
$$x_B = \begin{bmatrix} 1 & +\delta/\sqrt{d} & \cdots & +\delta/\sqrt{d} \end{bmatrix}^\top.$$

- The same part of the noise $\frac{\varepsilon_B - \varepsilon_A}{2}$ needs to be explained by a vector:

$$\frac{x_B - x_A}{2} = \frac{\delta}{\sqrt{d}} \cdot \begin{bmatrix} 0 & 1 & \cdots & 1 \end{bmatrix}^\top,$$

  which has norm δ, as before.

Double descent from 2D example

Suppose instead that x_A and x_B were actually $(d+1)$-dimensional vectors:

$$x_A = \begin{bmatrix} 1 & -\delta/\sqrt{d} & \cdots & -\delta/\sqrt{d} \end{bmatrix}^\top$$
$$x_B = \begin{bmatrix} 1 & +\delta/\sqrt{d} & \cdots & +\delta/\sqrt{d} \end{bmatrix}^\top.$$

- The same part of the noise $\frac{\varepsilon_B - \varepsilon_A}{2}$ needs to be explained by a vector:

$$\frac{x_B - x_A}{2} = \frac{\delta}{\sqrt{d}} \cdot \begin{bmatrix} 0 & 1 & \cdots & 1 \end{bmatrix}^\top,$$

which has norm δ, as before. But, the same noise is spread out across d directions:

$$\hat{\beta}_j = \frac{1}{\delta d} \frac{\varepsilon_B - \varepsilon_A}{2} \qquad j > 1.$$

Double descent from 2D example

We can now compute the generalization error:

$$\mathbb{E}\left[\left\|\hat{\beta}-\beta\right\|^2\right]=\mathbb{E}\left[\left(\frac{\varepsilon_A+\varepsilon_B}{2}\right)^2\right]+\sum_{j>1}\mathbb{E}\left[\left(\frac{1}{\delta d}\frac{\varepsilon_B-\varepsilon_A}{2}\right)\right]=\frac{1}{2}+\frac{1}{d}\frac{1}{2\delta^2}.$$

Double descent from 2D example

We can now compute the generalization error:

$$\mathbb{E}\left[\left\|\hat{\beta}-\beta\right\|^2\right]=\mathbb{E}\left[\left(\frac{\varepsilon_A+\varepsilon_B}{2}\right)^2\right]+\sum_{j>1}\mathbb{E}\left[\left(\frac{1}{\delta d}\frac{\varepsilon_B-\varepsilon_A}{2}\right)\right]=\frac{1}{2}+\frac{1}{d}\frac{1}{2\delta^2}.$$

- The $\frac{1}{2}$ terms comes from the noise explained by the first term.

Double descent from 2D example

We can now compute the generalization error:

$$\mathbb{E}\left[\left\|\hat{\beta}-\beta\right\|^2\right] = \mathbb{E}\left[\left(\frac{\varepsilon_A+\varepsilon_B}{2}\right)^2\right] + \sum_{j>1}\mathbb{E}\left[\left(\frac{1}{\delta d}\frac{\varepsilon_B-\varepsilon_A}{2}\right)\right] = \frac{1}{2}+\frac{1}{d}\frac{1}{2\delta^2}.$$

- The $\frac{1}{2}$ terms comes from the noise explained by the first term.
- The $\frac{1}{d}\frac{1}{2\delta^2}$ goes to zero as d goes to infinity.

Double descent in ordinary least squares: Gaussian model

Linear regression problem

Problem. Suppose nature generates data as follows:

$$y = x^\top \beta + \varepsilon,$$

- the covariates $x \in \mathbb{R}^d$ are d dimensional standard Gaussians, $x \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}_d\right)$
- the noise $\varepsilon \in \mathbb{R}$ is drawn from $\mathcal{N}(0, \sigma^2)$
- there is a true regressor β, but it is unknown to the learner

Goal. The goal of the learner is to use data to give an estimate $\hat{\beta}$ of β minimizing:

$$\|\hat{\beta} - \beta\|^2.$$

Linear regression on a single data point

Data. Let $y = x^\top \beta + \varepsilon$ where:

$$x \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}_d\right) \qquad \text{and} \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2).$$

Linear regression on a single data point

Data. Let $y = x^\top \beta + \varepsilon$ where:

$$x \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}_d\right) \qquad \text{and} \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2).$$

Solution. OLS returns the following:

$$\hat{\beta} = \frac{xy}{\|x\|^2}$$

Linear regression on a single data point

Data. Let $y = x^\top \beta + \varepsilon$ where:

$$x \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}_d\right) \qquad \text{and} \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2).$$

Solution. OLS returns the following:

$$\hat{\beta} = \frac{xy}{\|x\|^2} = \frac{xx^\top \beta + x\varepsilon}{\|x\|^2}$$

Linear regression on a single data point

Data. Let $y = x^\top \beta + \varepsilon$ where:

$$x \sim \mathcal{N}\Big(0, \frac{1}{d}\mathbf{I}_d\Big) \qquad \text{and} \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2).$$

Solution. OLS returns the following:

$$\hat{\beta} = \frac{xy}{\|x\|^2} = \frac{xx^\top \beta + x\varepsilon}{\|x\|^2} = \Pi_x \beta + \frac{x\varepsilon}{\|x\|^2},$$

where Π_x is the projection operator onto $\mathrm{span}(x)$.

Linear regression on a single data point

Data. Let $y = x^\top \beta + \varepsilon$ where:

$$x \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}_d\right) \qquad \text{and} \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2).$$

Solution. OLS returns the following:

$$\hat{\beta} = \frac{xy}{\|x\|^2} = \frac{xx^\top \beta + x\varepsilon}{\|x\|^2} = \Pi_x \beta + \frac{x\varepsilon}{\|x\|^2},$$

where Π_x is the projection operator onto $\mathrm{span}(x)$.

- Note that $\hat{\beta}$ satisfies: $x^\top \hat{\beta} = \dfrac{x^\top xy}{\|x\|^2} = y$

Generalization error

Denote $\beta_x = \Pi_x \beta$ and $\beta_x^\perp = \beta - \beta_x$ its orthogonal complement.

Generalization error

Denote $\beta_x = \Pi_x \beta$ and $\beta_x^\perp = \beta - \beta_x$ its orthogonal complement.

$$\mathbb{E}\left[\|\hat{\beta} - \beta\|^2\right]$$

Generalization error

Denote $\beta_x = \Pi_x \beta$ and $\beta_x^\perp = \beta - \beta_x$ its orthogonal complement.

$$\mathbb{E}\left[\|\hat{\beta} - \beta\|^2\right] = \mathbb{E}\left[\left\|\Pi_x \beta + \frac{x\varepsilon}{\|x\|^2} - \beta\right\|^2\right]$$

Generalization error

Denote $\beta_x = \Pi_x \beta$ and $\beta_x^\perp = \beta - \beta_x$ its orthogonal complement.

$$\mathbb{E}\left[\|\hat{\beta} - \beta\|^2\right] = \mathbb{E}\left[\left\|\Pi_x \beta + \frac{x\varepsilon}{\|x\|^2} - \beta\right\|^2\right]$$
$$= \mathbb{E}\left[\left\|\beta_x^\perp + \frac{x\varepsilon}{\|x\|^2}\right\|^2\right]$$

Generalization error

Denote $\beta_x = \Pi_x \beta$ and $\beta_x^\perp = \beta - \beta_x$ its orthogonal complement.

$$\begin{aligned}
\mathbb{E}\left[\|\hat{\beta} - \beta\|^2\right] &= \mathbb{E}\left[\left\|\Pi_x \beta + \frac{x\varepsilon}{\|x\|^2} - \beta\right\|^2\right] \\
&= \mathbb{E}\left[\left\|\beta_x^\perp + \frac{x\varepsilon}{\|x\|^2}\right\|^2\right] \\
&= \underbrace{\mathbb{E}\left[\|\beta_x^\perp\|^2\right]}_{\text{error from unseen directions}} + \underbrace{\mathbb{E}\left[\frac{\varepsilon^2}{\|x\|^2}\right]}_{\text{error from explaining noise}} .
\end{aligned}$$

Generalization error

Denote $\beta_x = \Pi_x \beta$ and $\beta_x^\perp = \beta - \beta_x$ its orthogonal complement.

$$
\begin{aligned}
\mathbb{E}\left[\|\hat{\beta} - \beta\|^2\right] &= \mathbb{E}\left[\left\|\Pi_x \beta + \frac{x\varepsilon}{\|x\|^2} - \beta\right\|^2\right] \\
&= \mathbb{E}\left[\left\|\beta_x^\perp + \frac{x\varepsilon}{\|x\|^2}\right\|^2\right] \\
&= \underbrace{\mathbb{E}\left[\|\beta_x^\perp\|^2\right]}_{\text{error from unseen directions}} + \underbrace{\mathbb{E}\left[\frac{\varepsilon^2}{\|x\|^2}\right]}_{\text{error from explaining noise}} .
\end{aligned}
$$

Notice that the last term is related to the signal-to-noise ratio.

Generalization error

Because x is isotropic Gaussian, the error from unseen directions is:

$$\underbrace{\mathbb{E}\left[\|\beta_x^{\perp}\|^2\right]}_{\text{error from unseen directions}} = \frac{d}{d-1}\|\beta\|^2.$$

Generalization error

Because x is isotropic Gaussian, the error from unseen directions is:

$$\underbrace{\mathbb{E}\left[\|\beta_x^\perp\|^2\right]}_{\text{error from unseen directions}} = \frac{d}{d-1}\|\beta\|^2.$$

And the error from explaining all the noise in the x direction:

$$\underbrace{\mathbb{E}\left[\frac{\varepsilon^2}{\|x\|^2}\right]}_{\text{error from explaining noise}} = \frac{\sigma^2}{d-2}$$

Generalization error

Because x is isotropic Gaussian, the error from unseen directions is:

$$\underbrace{\mathbb{E}\left[\|\beta_x^\perp\|^2\right]}_{\text{error from unseen directions}} \quad = \quad \frac{d}{d-1}\|\beta\|^2.$$

And the error from explaining all the noise in the x direction:

$$\underbrace{\mathbb{E}\left[\frac{\varepsilon^2}{\|x\|^2}\right]}_{\text{error from explaining noise}} \quad = \quad \frac{\sigma^2}{d-2}$$

- Note that when x is standard normal, $1/\|x\|^2$ follows an **inverse Wishart distribution** (and $\|x\|^2$ follows a χ^2-distribution with degree of freedom d).

Generalization error

Because x is isotropic Gaussian, the error from unseen directions is:

$$\underbrace{\mathbb{E}\left[\|\beta_x^{\perp}\|^2\right]}_{\text{error from unseen directions}} \quad = \quad \frac{d}{d-1}\|\beta\|^2.$$

And the error from explaining all the noise in the x direction:

$$\underbrace{\mathbb{E}\left[\frac{\varepsilon^2}{\|x\|^2}\right]}_{\text{error from explaining noise}} \quad = \quad \frac{\sigma^2}{d-2}$$

- Note that when x is standard normal, $1/\|x\|^2$ follows an **inverse Wishart distribution** (and $\|x\|^2$ follows a χ^2-distribution with degree of freedom d).
- If $d = 1$ or $d = 2$, then the expected generalization error is infinite.

χ^2-distribution

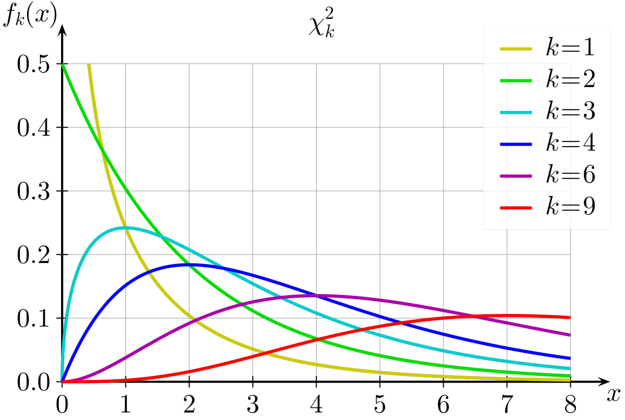


Figure 6: χ^2-distributions where k is the degree of freedom, from *Wikipedia*.

Main result from Belkin et al. (2020)

Setting. Their setting extends this setting of linear regression on single point $x \in \mathbb{R}^d$.

Main result from Belkin et al. (2020)

Setting. Their setting extends this setting of linear regression on single point $x \in \mathbb{R}^d$.

- They train a regressor on n data points $x_1, \ldots, x_n \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}_d\right)$.

Main result from Belkin et al. (2020)

Setting. Their setting extends this setting of linear regression on single point $x \in \mathbb{R}^d$.

- They train a regressor on n data points $x_1, \ldots, x_n \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}_d\right)$.
- To compare across size of model, only p random dimensions of $\mathbb{R}^d$ are revealed.

Main result from Belkin et al. (2020)

Setting. Their setting extends this setting of linear regression on single point $x \in \mathbb{R}^d$.

- They train a regressor on n data points $x_1, \ldots, x_n \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}_d\right)$.
- To compare across size of model, only p random dimensions of $\mathbb{R}^d$ are revealed.

Theorem (Belkin et al. (2020))

Let $\hat{\beta}$ be the OLS regressor in this setting. Then its expected risk is:

$$\mathbb{E}\left[\left(y - x^\top \hat{\beta}\right)^2\right] = \begin{cases} \left(\left(1 - \frac{p}{d}\right) \cdot \|\beta\|^2 + \sigma^2\right) \cdot \left(1 + \frac{p}{n-p-1}\right) & p \leq n-2 \\ \infty & p = n, n+1 \\ \|\beta\|^2 \cdot \left(1 - \frac{n}{d} \cdot \left(2 - \frac{d-n-1}{p-n-1}\right)\right) + \sigma^2 \cdot \left(1 + \frac{n}{p-n-1}\right) & p \geq n+2 \end{cases}$$

Main result from Belkin et al. (2020)

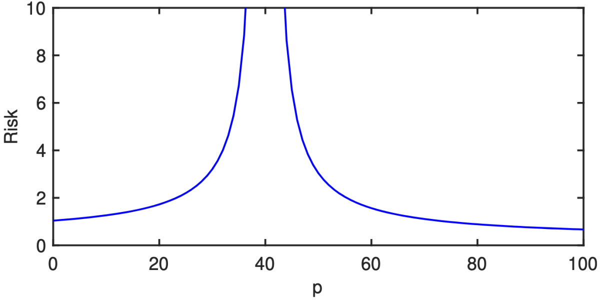


Figure 7: Visualization of the double descent curve from previous theorem(Belkin et al., 2020).

References

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine learning practice and the bias-variance trade-off. *arXiv preprint arXiv:1812.11118*, 2018.

Mikhail Belkin, Daniel Hsu, and Ji Xu. Two models of double descent for weak features. *SIAM Journal on Mathematics of Data Science*, 2(4):1167–1180, 2020.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003, 2021.

M Opper, W Kinzel, J Kleinz, and R Nehl. On the ability of the optimal perceptron to generalise. *Journal of Physics A: Mathematical and General*, 23(11):L581, 1990.