Resampling Methods

Last modified: 2026-11-19

> **Source**
>
> Adapted from Kameron Decker Harris' 2022 weekly notes.

1 Resampling methods

Resampling methods draw repeated samples[1] from the data we have, and fit the same model again to each one. Comparing the fits shows what a single fit cannot, such as how well the model predicts new data.

2 Holding out a validation set

The simplest approach splits the data once, into the three parts in Figure 1. We fit each candidate model on the training set, choose the candidate with the smallest validation error, and report the chosen model's test error.



Figure 1: One rectangle cut into three parts, left to right: a wide part labeled Train, a narrower part labeled Val, and a narrow part labeled Test.

With only one candidate, nothing is chosen. A validation set is then not needed, and the two-part split in Figure 2 suffices.

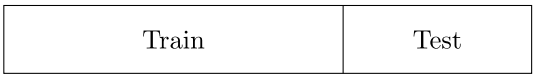


Figure 2: One rectangle cut into two parts: a wide part labeled Train on the left and a narrower part labeled Test on the right.

2.1 Weighing a single split

A single split assumes that the observations[2] are independent and identically distributed[3]. Even then, it has three drawbacks:

- The held-out parts shrink the training set.
- The validation error depends on which observations the random[4] split puts in the validation set.

[1] https://morrison-lab.github.io/sds/exploratory-descriptive.html#def-sample
[2] https://morrison-lab.github.io/sds/exploratory-descriptive.html#def-observation
[3] https://morrison-lab.github.io/pds/independence.html#def-iid
[4] https://morrison-lab.github.io/pds/notation.html#def-random

- As an estimate[5] of the chosen model's risk, the validation error can therefore have a high variance[6].

> **Watch**
>
> Hastie and Tibshirani describe the validation-set approach, from 8:11[a], and its drawbacks, from 13:14[b] [1], video 5.1.
>
> [a] https://www.youtube.com/watch?v=6eWODQJrMKs&t=491s
> [b] https://www.youtube.com/watch?v=6eWODQJrMKs&t=794s

> **Watch**
>
> Abu-Mostafa covers model selection with a validation set in his lecture on validation, from 30:47[a] [2], Lecture 13.
>
> [a] https://www.youtube.com/watch?v=o7zzaKd0Lkk&t=1847s

> **Watch**
>
> The CS229 lecture covers data splits, models and cross-validation[a] [3], Lecture 8.
>
> [a] https://www.youtube.com/watch?v=rjbkWSTjHzM

> **Watch**
>
> Lall compares validation approaches in his EE104 lecture on validation, from 14:20[a] [4], Lecture 4.
>
> [a] https://www.youtube.com/watch?v=N2lwsB1qfJw&t=860s

3 K-fold cross-validation

K-fold cross-validation splits the data into K folds. This K is not the number of neighbors in K-nearest neighbors. Both names are standard, so the letter is reused. Figure 3 shows $K = 4$.

[5] https://morrison-lab.github.io/sds/estimation.html#def-estimate
[6] https://morrison-lab.github.io/pds/variance-covariance.html#def-variance

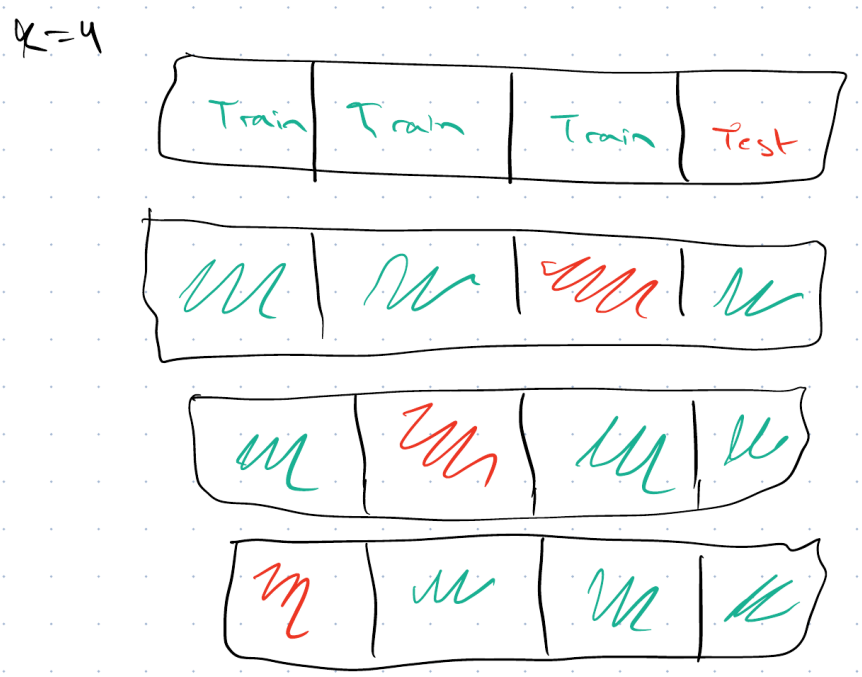


Figure 3: Four bars, each cut into four folds, for $K = 4$. The last fold of the first bar is the test fold. In each of the other three bars a different one of the first three folds is marked in red as the test fold, and the other three are marked in green.

The test fold rotates through the K folds. Each turn fits the model on the other $K - 1$ folds and scores it on the test fold. The K-fold cross-validation estimate of the risk combines the K scores. When the folds have the same size, the cross-validation estimate is the sample mean[7] of the K scores. Every observation[8] is scored once, so the cross-validation estimate varies less than the validation error of a single split. The price is K fits instead of one.

Figure 4 runs K-fold cross-validation on the Auto data, one fold at a time. The mean squared errors of the folds vary from fold to fold, and their sample mean is the K-fold cross-validation estimate.

```
kr_static_points |>
  ggplot2::ggplot() +
  ggplot2::aes(x = horsepower, y = mpg) +
  ggplot2::geom_point(ggplot2::aes(colour = role), size = 0.5) +
  ggplot2::geom_line(data = kr_static_curves, linewidth = 0.7) +
  ggplot2::geom_text(
    data = kr_static_labels, ggplot2::aes(label = label),
    x = 230, y = 47, hjust = 1, size = 2.8
  ) +
  ggplot2::facet_wrap(ggplot2::vars(fold), nrow = 1, labeller = "label_both") +
  ggplot2::scale_colour_manual(values = kr_colours) +
  ggplot2::coord_cartesian(ylim = kr_ylim) +
  ggplot2::labs(colour = NULL) +
  ggplot2::theme(legend.position = "bottom")
```

[7] https://morrison-lab.github.io/sds/exploratory-descriptive.html#def-sample-mean
[8] https://morrison-lab.github.io/sds/exploratory-descriptive.html#def-observation

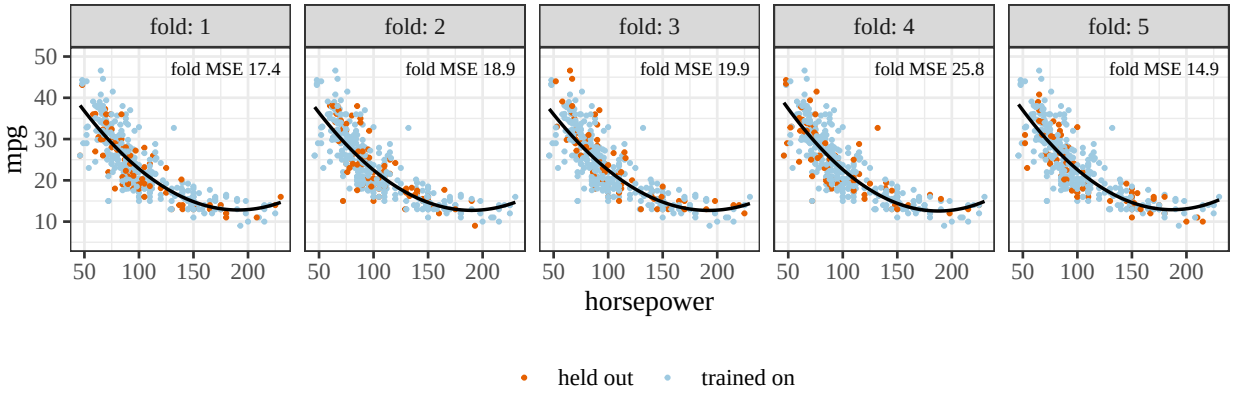


Figure 4: K-fold cross-validation of a polynomial[9] regression of `mpg` on `horsepower`, on the 392 cars of the Auto data, one held-out fold at a time. Each fit trains on the other $K-1$ folds, drawn in light blue, and is scored on the held-out fold, drawn in orange. The website lets you choose K, the held-out fold and the degree[10], and plots the MSE on every held-out fold, with a dashed line at their sample mean[11], the K-fold estimate[12]. The handout shows the 5 folds of a degree-2 fit, whose 5-fold estimate is 19.4.

3.1 Leave-one-out cross-validation

Leave-one-out cross-validation is the extreme case $K = n$, in which each fold holds a single observation[13].

```
vv_all |>
  ggplot2::ggplot() +
  ggplot2::aes(x = degree, y = mse, group = rep, colour = rep_label) +
  ggplot2::geom_line() +
  ggplot2::geom_point(size = 0.8) +
  ggplot2::facet_wrap(ggplot2::vars(method)) +
  ggplot2::scale_x_continuous(breaks = vv_degrees) +
  ggplot2::coord_cartesian(ylim = vv_ylim) +
  ggplot2::labs(
    x = "polynomial degree", y = "estimated test MSE (mpg^2)"
  ) +
  ggplot2::theme(legend.position = "none")
```

[9] https://morrison-lab.github.io/mds/algebra-exponentials.html#def-polynomial
[10] https://morrison-lab.github.io/mds/algebra-exponentials.html#def-polynomial-degree
[11] https://morrison-lab.github.io/sds/exploratory-descriptive.html#def-sample-mean
[12] https://morrison-lab.github.io/sds/estimation.html#def-estimate
[13] https://morrison-lab.github.io/sds/exploratory-descriptive.html#def-observation

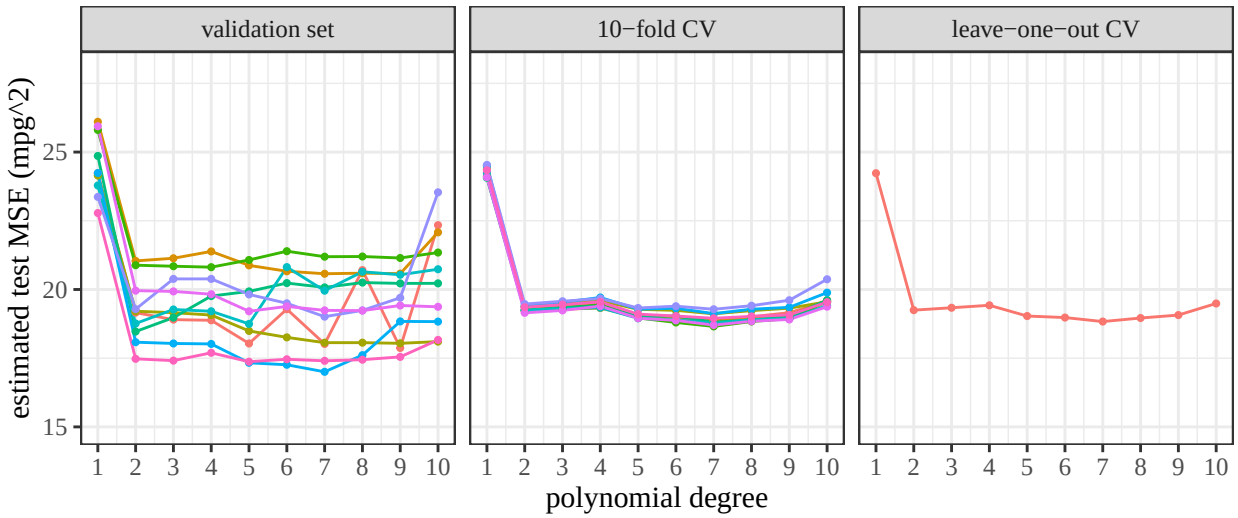


Figure 5: Three estimates[14] of the test mean squared error of a polynomial[15] regression of `mpg` on `horsepower`, for the 392 cars of the Auto data, by degree[16]. Each validation-set line comes from a different random[17] half-and-half split, and each 10-fold line comes from a different random assignment of the cars to folds. Leave-one-out has one line, because it involves no randomness. The handout shows all 10 repetitions; the website adds them one at a time with the slider. The analysis follows Figures 5.2 and 5.4 of *An Introduction to Statistical Learning* [5].

Figure 5 compares three estimates[18] of the test mean squared error on the Auto data. For the degree-2 fit, the ten estimates from a random[19] validation set run from 17.5 to 21, the ten 10-fold estimates run from 19.2 to 19.5, and leave-one-out cross-validation gives 19.2. Averaging over folds makes the 10-fold estimate far less variable than a single split.

Watch

Abu-Mostafa explains leave-one-out cross-validation, from 49:03[a] [2], Lecture 13.

[a] https://www.youtube.com/watch?v=o7zzaKd0Lkk&t=2943s

Watch

Hastie and Tibshirani go through K-fold cross-validation in detail, from 0:55[a] [1], video 5.2.

[a] https://www.youtube.com/watch?v=AMfvd_hLssE&t=55s

Watch

Starmer explains K-fold cross-validation[a] [6].

[a] https://www.youtube.com/watch?v=fSytzGwwBVw

[14] https://morrison-lab.github.io/sds/estimation.html#def-estimate
[15] https://morrison-lab.github.io/mds/algebra-exponentials.html#def-polynomial
[16] https://morrison-lab.github.io/mds/algebra-exponentials.html#def-polynomial-degree
[17] https://morrison-lab.github.io/pds/notation.html#def-random
[18] https://morrison-lab.github.io/sds/estimation.html#def-estimate
[19] https://morrison-lab.github.io/pds/notation.html#def-random

Watch

Abu-Mostafa explains K-fold cross-validation, from 1:06:25[a] [2], Lecture 13.

[a] https://www.youtube.com/watch?v=o7zzaKd0Lkk&t=3985s

Watch

Lall covers cross-validation in his EE104 lecture on validation, from 36:24[a] [4], Lecture 4.

[a] https://www.youtube.com/watch?v=N2lwsB1qfJw&t=2184s

Read

GeeksforGeeks explains cross-validation[a] [7], including K-fold and leave-one-out.

[a] https://www.geeksforgeeks.org/machine-learning/cross-validation-machine-learning/

Read

The University of Washington's CSE 446 slides cover train-test splits and cross-validation[a] [8], Lecture 5.

[a] https://courses.cs.washington.edu/courses/cse446/26sp/schedule/lectures/05-train%20test%20split%20cross%20validation.pdf

[1] T. Hastie, R. Tibshirani, and J. Taylor, "Statistical learning with python." Stanford Online, lecture videos on YouTube, 2023. Available: https://www.youtube.com/playlist?list=PLoROMvodv4rPP6braWoRt5UCXYZ71GZIQ

[2] Y. S. Abu-Mostafa, "Learning from data: CS 156 lectures." Caltech, lecture videos on YouTube, 2012. Available: https://www.youtube.com/playlist?list=PLD63A284B7615313A

[3] A. Ng *et al.*, "CS229: Machine learning, autumn 2018." Stanford Online, lecture videos on YouTube, 2018. Available: https://www.youtube.com/playlist?list=PLoROMvodv4rMiGQp3WXShtMGgzqpfVfbU

[4] S. Lall, "EE104: Introduction to machine learning, 2020." Stanford Online, lecture videos on YouTube, 2020. Available: https://www.youtube.com/playlist?list=PLoROMvodv4rN_Uy7_wmS051_q1d6akXmK

[5] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning with applications in R*, 2nd ed. Springer, 2021. Available: https://www.statlearning.com/

[6] J. Starmer, "Machine learning fundamentals: Cross validation." StatQuest, video on YouTube, 2018. Available: https://www.youtube.com/watch?v=fSytzGwwBVw

[7] GeeksforGeeks, "Cross validation in machine learning." GeeksforGeeks, tutorial article, 2026. Available: https://www.geeksforgeeks.org/machine-learning/cross-validation-machine-learning/

[8] N. Jaques, "CSE 446/546: Machine learning, lecture slides." University of Washington, 2026. Accessed: Oct. 12, 2026. [Online]. Available: https://courses.cs.washington.edu/courses/cse446/26sp/schedule/lectures/