# Randomization

## Practical Considerations

Make your data generation pipeline reproducible. Say you want to add a feature to see how it affects model quality. For a fair experiment, your datasets should be identical except for this new feature. If your data generation runs are not reproducible, you can't make these datasets.

In that spirit, make sure any randomization in data generation can be made deterministic:

- **Seed your random number generators** (RNGs). Seeding ensures that the RNG outputs the same values in the same order each time you run it, recreating your dataset.
- **Use invariant hash keys.** Hashing (https://wikipedia.org/wiki/Hash_function) is a common way to split or sample data. You can hash each example, and use the resulting integer to decide in which split to place the example. The inputs to your hash function shouldn't change each time you run the data generation program. Don't use the current time or a random number in your hash, for example, if you want to recreate your hashes on demand.

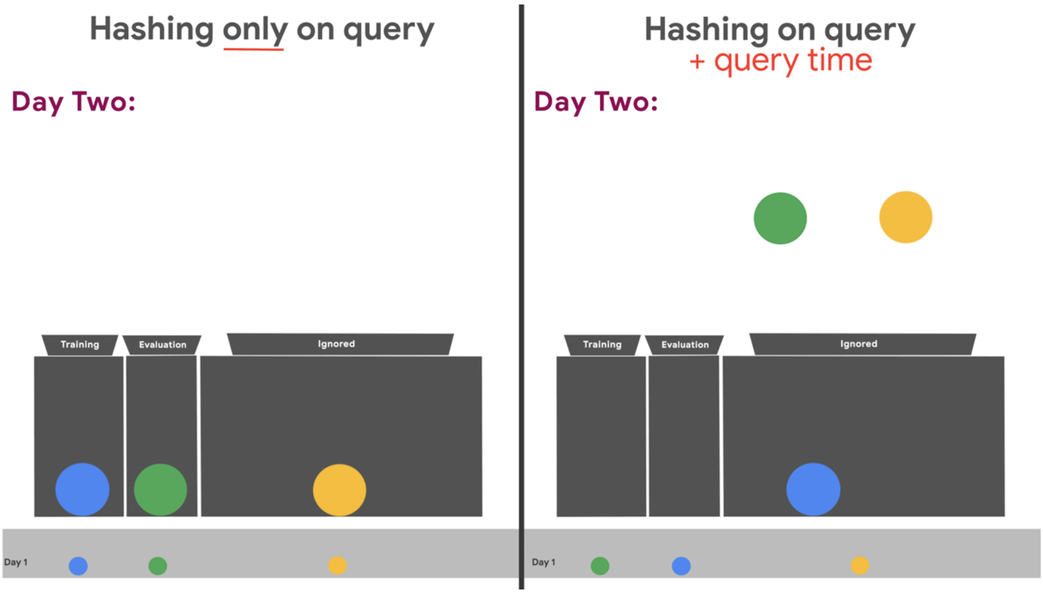
The preceding approaches apply both to sampling and splitting your data.

## Considerations for Hashing

Imagine again you were collecting Search queries and using hashing to include or exclude queries. If the hash key only used the query, then across multiple days of data, you'll either *always* include that query or *always* exclude it. Always including or always excluding a query is bad because:

- Your training set will see a less diverse set of queries.
- Your evaluation sets will be artificially hard, because they won't overlap with your training data. In reality, at serving time, you'll have seen some of the live traffic in your training data, so your evaluation should reflect that.

Instead you can hash on query + date, which would result in a different hashing each day.



your hashing unique to ensure your system doesn't collide with other systems.

Previous
← Splitting Your Data (/machine-learning/data-prep/construct/sampling-splitting/split)

Next
Check Your Understanding → (/machine-learning/data-prep/construct/sampling-splitting/check-your-understanding)

Last updated 2019-07-11 UTC.