**Validación de nuestro modelo usando Cross Validation**

1. **La última palabra siempre la van a tener los datos.**

* Todas nuestras intuiciones no tiene nada que hacer frente a lo que digan los datos y las matemáticas que aplicamos sobre estos datos. Por eso es importante siempre tener rigurosidad a la hora de evaluar los resultados que estamos recibiendo.

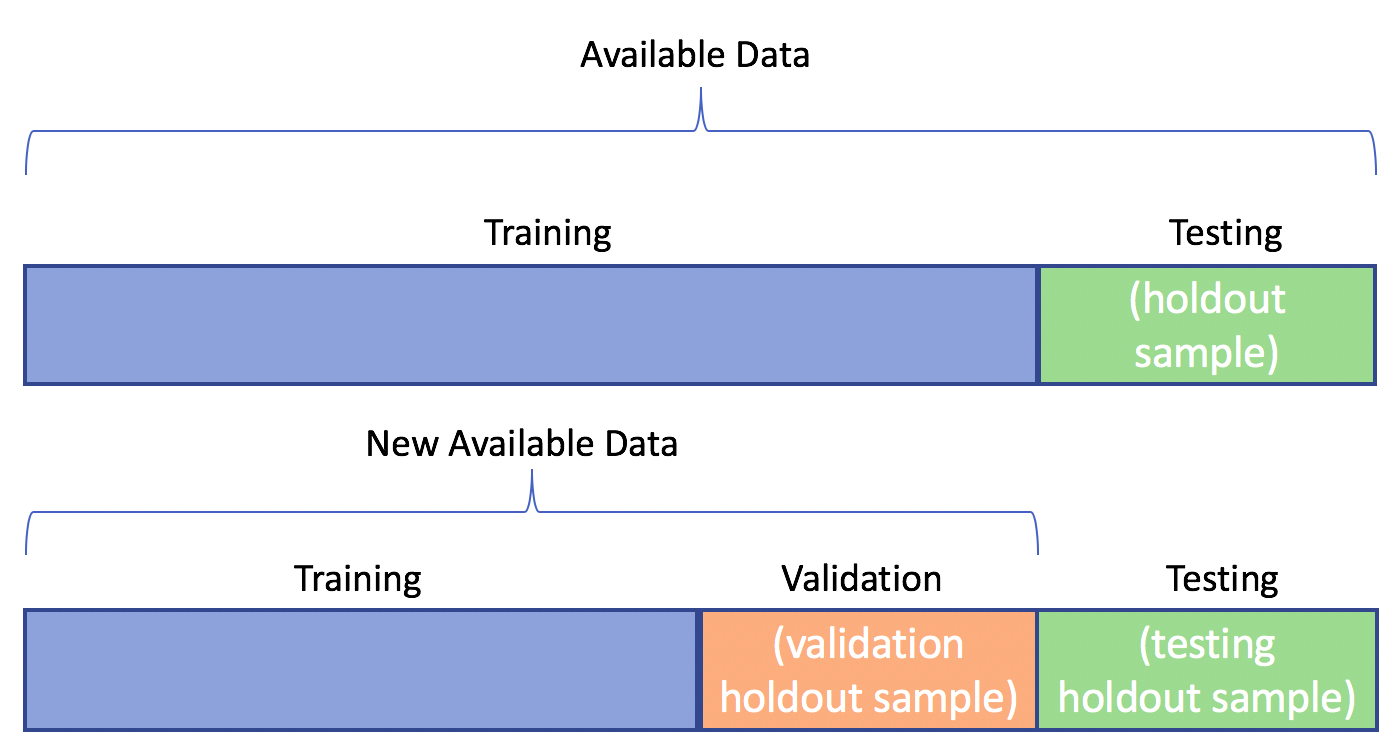
1. **Necesitamos mentalidad de testeo.**

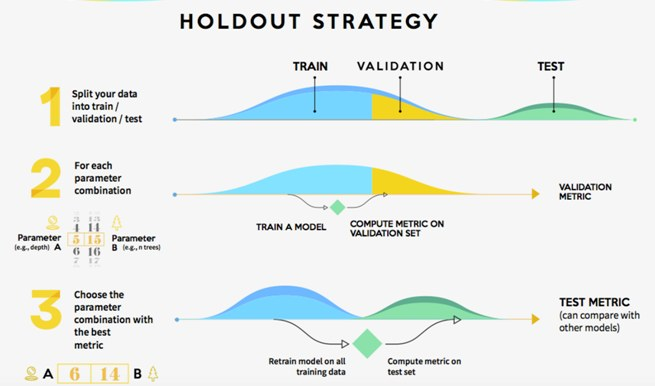
* No se trata solamente de probar un poco al principio y un poco al final, sino que tendremos que probar constantemente durante todo el proceso, para poder encontrar cuál es la solución óptima que realmente nos soluciona el problema que tenemos pendiente, todo esto:
  + con varias formas
  + con varios conjuntos de datos
  + con varias configuraciones de parámetros
  + con varias distribuciones de nuestros datos

1. **Todos los modelos son malos, solamente algunos son útiles.**

* Todos los modelos que nosotros hacemos en últimas son una sobre simplificación de lo que pasa realmente. Entonces nunca nuestros modelos van a corresponder con la realidad al cien por ciento. Si jugamos lo suficiente y si somos lo suficientemente hábiles para configurar, vamos a llegar a un punto donde el modelo que estamos trabajando va a ser útil para ciertos casos específicos dentro del mundo real.  
  .  
  .  
  .  
  **Tipos de validación**
* ***Hold-Out***

Se trata de dividir nuestros datos entrenamiento/pruebas, básicamente consiste en usar porcentajes fijos, por lo regular 70% de entrenamiento y 30% de pruebas.

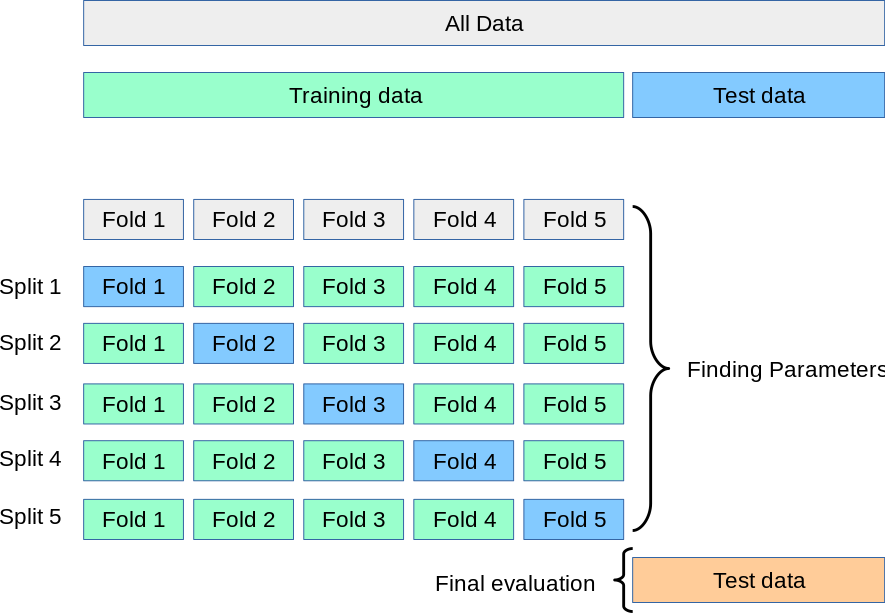


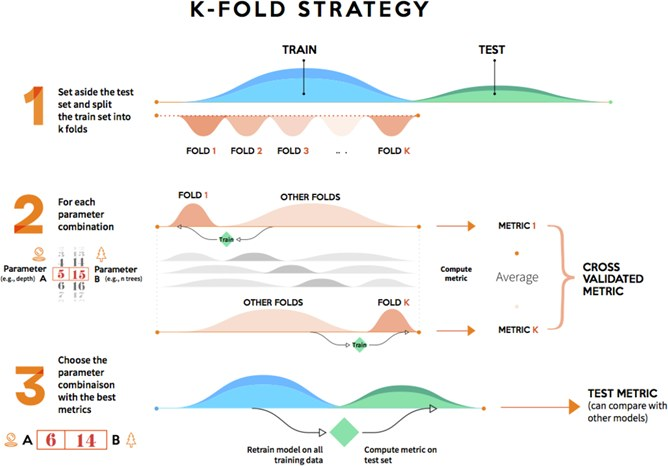


**¿Cuándo utilizar Hold-on?**

* Se requiere un prototipado rápido.
* No se tiene mucho conocimiento en ML.
* No se cuenta con abundante poder de cómputo.  
  .  
  .
* ***K-Folds***

Usar validación cursada K-Fold, aquí vamos a plegar nuestros datos k veces, el k es un parámetro que nosotros definimos y en esos pliegues vamos a utilizar diferentes partes de nuestro dataset como entrenamiento y como test, de tal manera que intentemos cubrir todos los datos de entrenamiento y de test, al finalizar el proceso.

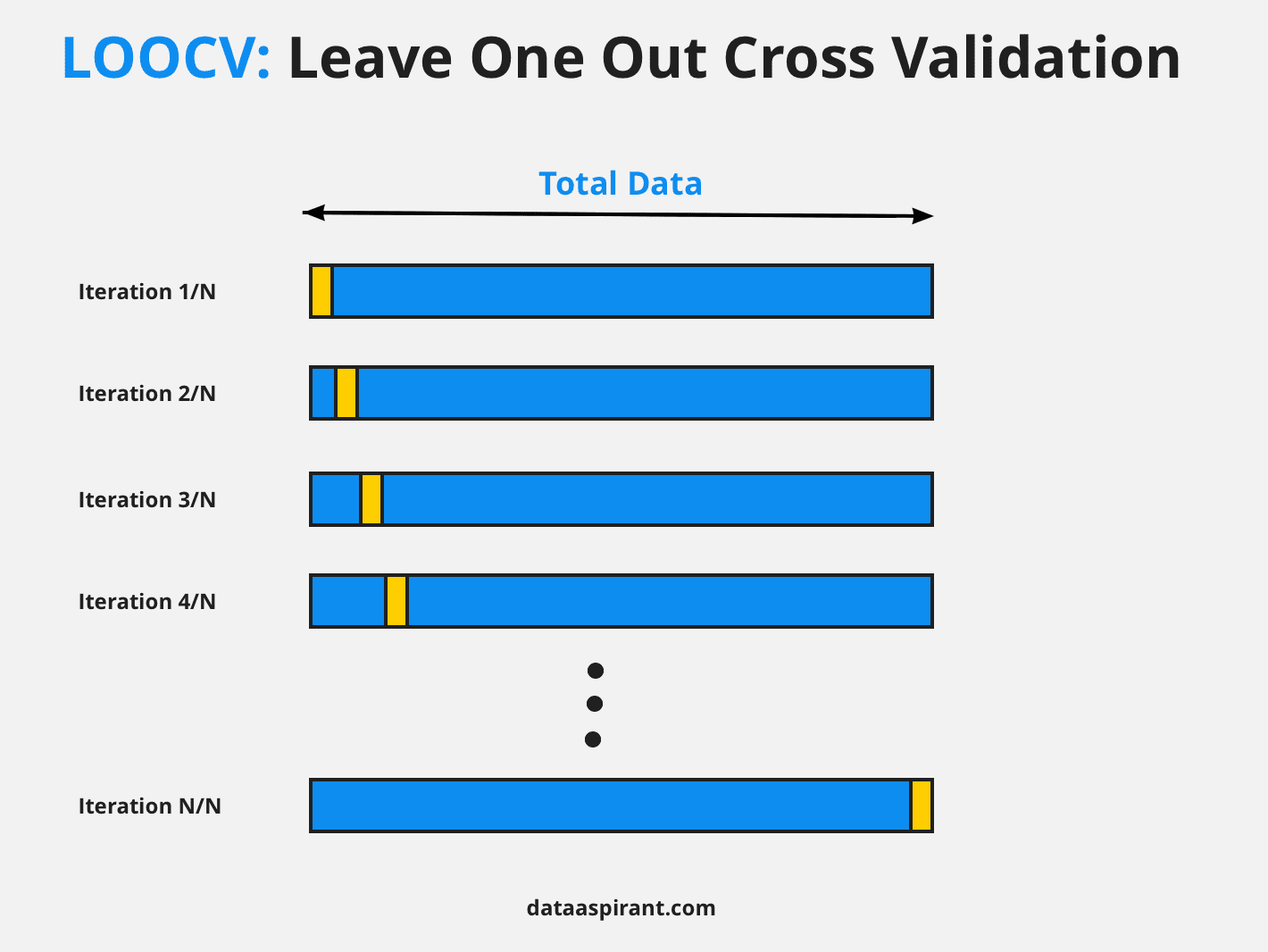




**¿Cuándo utilizar K-Folds?**

* Recomendable en la mayoría de los casos.
* Se cuenta con un equipo suficiente para desarrollar ML.
* Se require la integración con técnicas de optimización paramétrica.
* Se tiene más tiempo para las pruebas.  
  .  
  .
* ***LOOCV***

Validación cruzada LOOCV, Leave One Out Cross Validation. Este es el método más intensivo, ya que haremos una partición entre entrenamiento y pruebas, porque vamos a hacer entrenamiento con todos los datos, salvo 1 y vamos a repetir este proceso tantas veces hasta que todos los datos hayan sido probados.



**¿Cuándo utilizar LOOCV?**

* Se tiene gran poder de computo
* Se cuetan con pocos datos para poder dividir por Train/Test
* Cuando se quiere probar todos los casos posibles (para personas con TOC)