

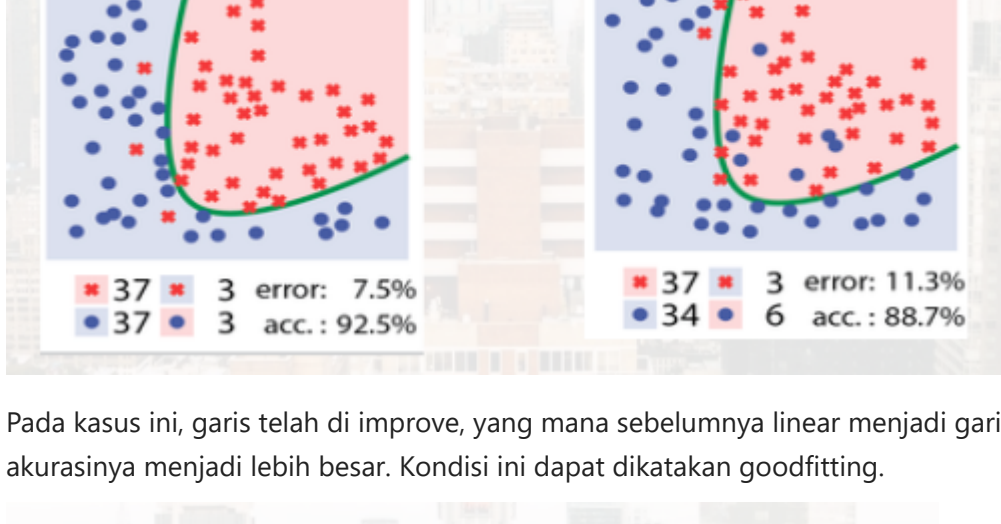
Model Validation

A basic method of evaluation is model validation on a different, independent data set from the same source, i.e. on testing data.

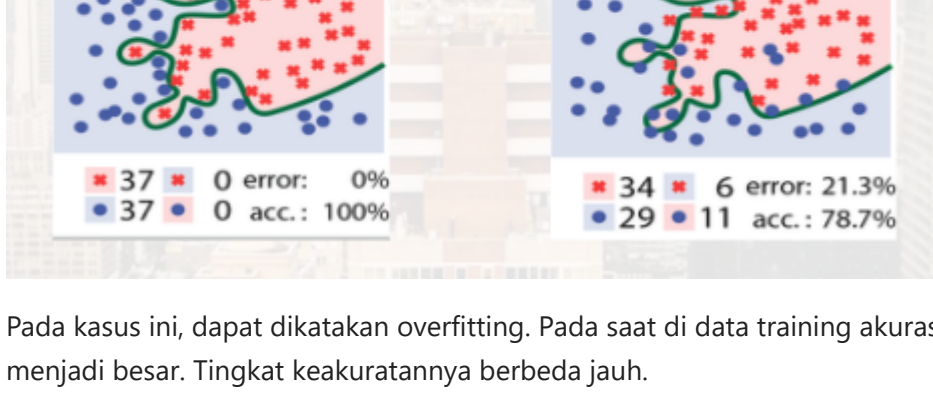
Model Validation adalah melakukan evaluasi model menggunakan data yang tidak digunakan pada saat training. Misalnya dalam training data, kita menggunakan data x1. Pada saat evaluasi model atau proses testing data, kita menggunakan data x2 (data yang berbeda).



- Pada contoh di atas, model klasifikasi terlalu sederhana (linear) yang mana masih bisa untuk ditingkatkan lagi. Jika tidak kondisi tersebut dikhawatirkan akan menyebabkan underfitting
- Catatan: Jika hasil di data training berbeda jauh dengan hasil di data testing dapat diindikasikan overfitting.



Pada kasus ini, garis telah di improve, yang mana sebelumnya linear menjadi garis kuadratik. Sehingga errornya menjadi lebih kecil dan akurasi menjadi lebih besar. Kondisi ini dapat dikatakan goodfitting.

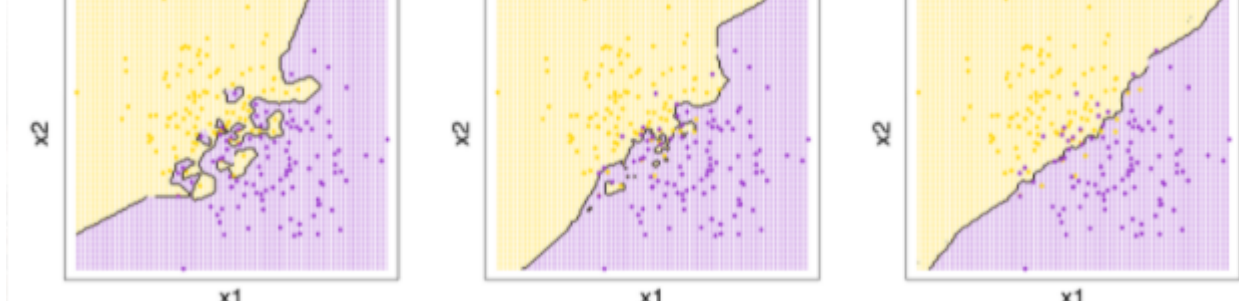


Pada kasus ini, dapat dikatakan overfitting. Pada saat di data training akurasi 100%. Namun pada saat di data testing maka errornya menjadi besar. Tingkat keakuratannya berbeda jauh.

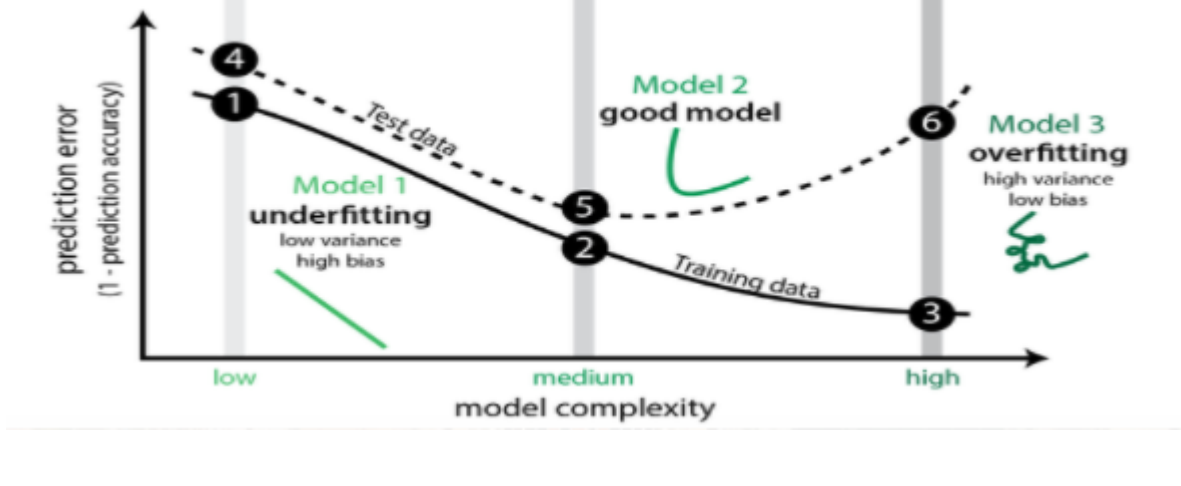
Becareful Underfitting and Overfitting

If the model is too simple, the solution is biased and does not fit the data.

If the model is too complex then it is very sensitive to small changes in the data.



Model Validation



- Nomor 1, 2, 3 merupakan data training.
- Nomor 4, 5, 6 merupakan data testing.

How to Prevent Underfitting and Overfitting?

Choose the Best Model

- Menggunakan data yang berbeda saat membuat model dan saat melakukan evaluasi model.

How to chose the Best Model?

Choose the Right Validation Method

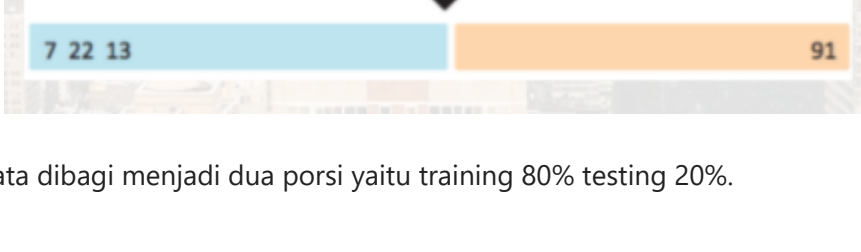
Type of Validation Method

- Train and Test Sets
- K-fold Cross Validation.
- Leave One Out Cross Validation.

Train and Test Sets

It involves randomly dividing the available dataset of observations into two parts.

- A training set



- A validation set or hold out set

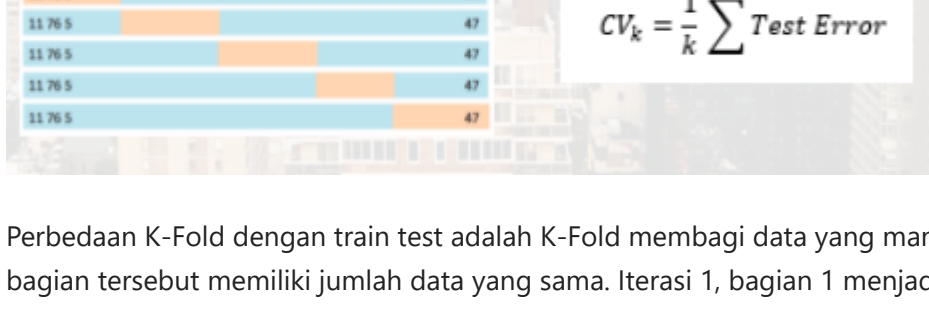
Misalnya memiliki 1000 data, lalu data dibagi menjadi dua porsi yaitu training 80% testing 20%.

k-Fold Cross Validations

This approach involves randomly dividing the set of observations into k groups, or fold, of approximately equal size.

The first fold is treated as validation set, and the method is fit on the remaining k-1 folds.

The k-fold CV estimate is computed by averaging these values



Perbedaan K-Fold dengan train test adalah K-Fold membagi data yang mana pembagian tersebut di iterasi. Misalnya memilih K = 5, 5 bagian tersebut memiliki jumlah data yang sama. Iterasi 1, bagian 1 menjadi data training dan 4 bagian lainnya menjadi data testing. Iterasi 2, bagian 2 menjadi data training dan 4 bagian lainnya menjadi data testing. dan seterusnya.

Tujuannya agar akurasi yang Robust atau sesuai.

Leave One Out Cross Validation

You can configure cross validation so that the size of the fold is 1 (k is set to the number of observations in your dataset). This variation of cross validation is called leave-one-out cross validation.

The result is a large number of performance measures that can be summarized in an effort to give a more reasonable estimate of the accuracy of your model on unseen data. A downside is that it can be a computationally more expensive procedure than k-fold cross



validation.

Sama seperti K-Fold, perbedaannya Leave One Out Cross Validation membagi data sebanyak jumlah data yang digunakan pada saat training. Misalnya jumlah data yang digunakan sebanyak 1000 maka data diiterasi sebanyak 1000.

What Techniques to Use When

Generally k-fold cross validation is the gold-standard for evaluating the performance of a machine learning algorithm on unseen data with k set to 3, 5, or 10.

Using a train/test split is good for speed when using a slow algorithm and produces performance estimates with lower bias when using large datasets.

Techniques like leave-one-out cross validation can be useful intermediates when trying to balance variance in the estimated performance, model training speed and dataset size.

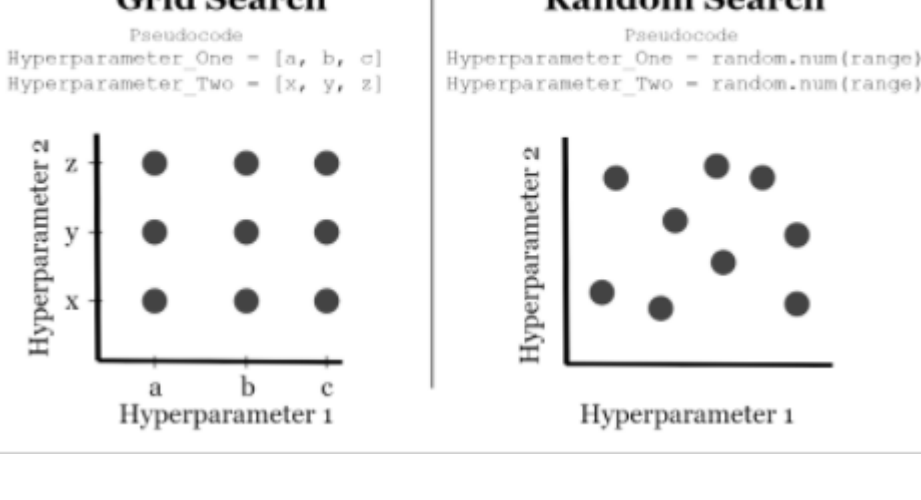
The best advice is to experiment and find a technique for your problem that is fast and produces reasonable estimates of performance that you can use to make decisions. If in doubt, use a combination of train/test split(80:20) and 10-fold cross validation.

Choose the Right Hyperparameter Model

Mengoptimalkan nilai-nilai yang terdapat di dalam model.

Tunning Hyperparameter

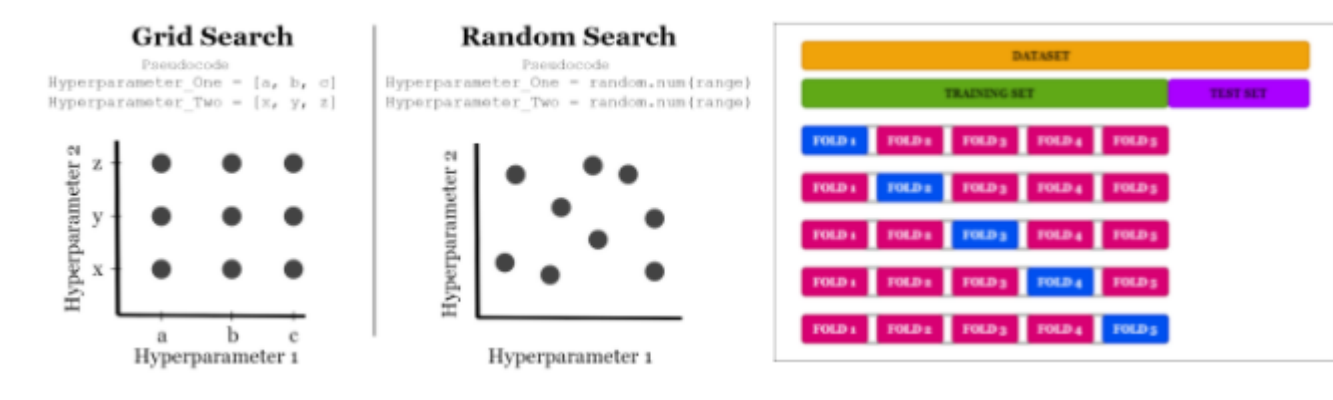
Hyperparameter tuning is choosing a set of optimal hyperparameters for a learning algorithm



Metode dalam melakukan Turning Hyperparameter:

- Grid Search akan mencari kombinasi di antara Hyperparameter 1 dan Hyperparameter 2 yang mana digunakan di dalam model. Misalnya kombinasi antara a dengan x, kombinasi antara a dengan y dan seterusnya. Dari 9 kombinasi, dicari kombinasi yang menghasilkan kombinasi terbaik
- Random Search: tidak semua kombinasi digunakan, hanya menggunakan kombinasi tertentu. Misalnya dari 9 kombinasi akan dirandom menjadi 5 kombinasi. Dan dari 5 kombinasi akan dipilih yang terbaik. Digunakan untuk mempersingkat proses komputasi.

Tunning Hyperparameter+CV



Digabungkan antara K-Fold dengan Grid search. Maka misalnya K = 5 dan kombinasinya 9 maka akan dilakukan iterasi sebanyak 9 x 5 = 45 kali.

Cara memiliki Hyperparameter optimal?

Di rata-ratakan. Dan kombinasi yang memiliki rata-rata terbesar akan dipilih.

Metrics Evaluation

What is The Good Measure of "Model Quality" from Machine Learning Standpoint?

Measure of Models Error

Accuracy

Accuracy is GOD

- Include Target Column as Features
- For classification : Only check global accuracy
- Use old algorithm

Accuracy Paradox

Recurrence of Breast Cancer

Total Observation : 286

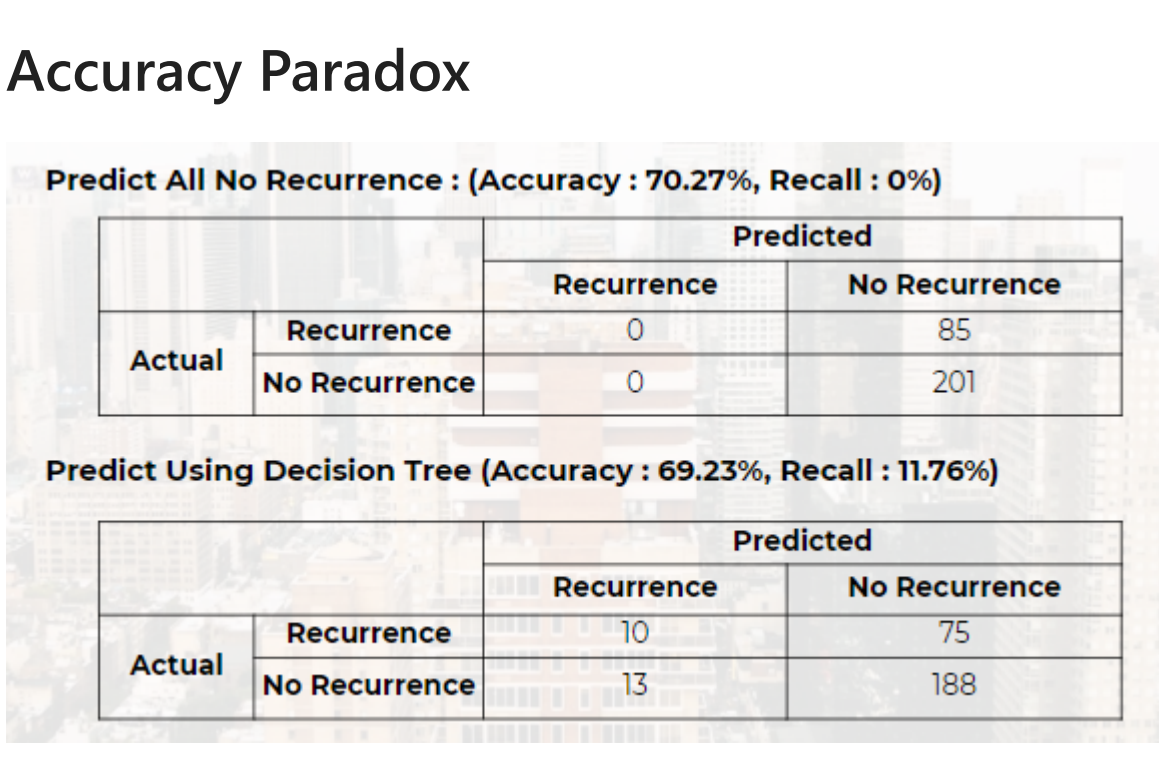
Actual (Imbalance Datasets)

Recurrence : 85

No Recurrence : 201

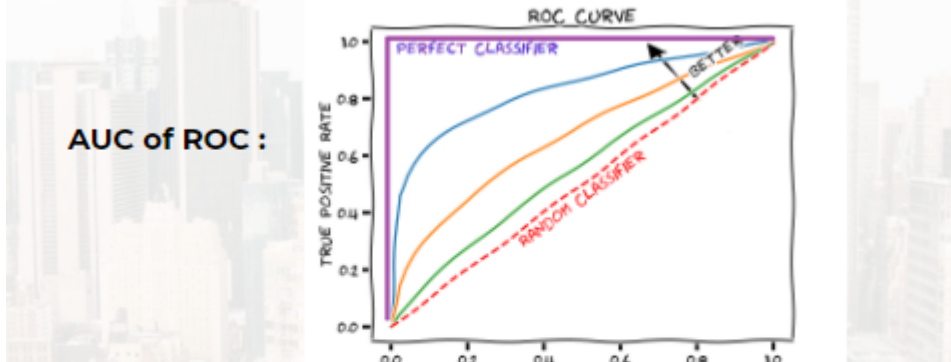
We want to build machine learning model to predict recurrence of breast cancer

Accuracy Paradox



Akurasi tinggi di bagian 1 tetapi recallnya lebih tinggi yang bagian 2. Mana yang lebih baik? Sebenarnya tidak ada yang bagus. Namun metode kedua lebih baik karena mampu memprediksi recurrence sedangkan metode pertama tidak mampu memprediksi recurrence.

Metrics for Imbalance Dataset



Jika hanya mengandalkan akurasi, maka model yang dibangun tidak bagus untuk data yang imbalance. Sehingga memerlukan metrics lain dalam mengevaluasi.

Terdapat 2 metrics:

- AUC OF RUC
- F1-Score: