Machine Learning Engineer Course
Day 8

- Machine Learning Flow -

DIVE INTO CODE

Thursday April 29th , 2021

DIOP Mouhamed

Agenda

1. Check-in
2. How to proceed
3. Quick Review
4. ML Flow
5. Evaluation of Prediction Accuracy
6. Confusion Matrix
7. Evaluation of the Model
8. Cross-Validation
9. Parameter tuning
10. Kaggle Submission

Check-in

3 minutes Please post the following point to Zoom chat.

Q. What do you want to know the most right now?
(Anything is fine.)

How to proceed - Objective

Purpose of learning. Purpose clarifies a person's role and the learning required. Clear learning leads to a sense of growth and confidence.

	Objective	NOT Objective
1	**Learn how to think about the program with your peers**	Memorize lots of functions
2	**Use the basic elements of the program**	Complete assignments quickly
3	**Feel like a fresh business person**	

How to proceed - Objective

How to solve problems
“Sprint machine learning flow”

Using Kaggle's Home Credit Default Risk

Half will be explained by me. Do the other half yourself.

[Problem 1] Cross Validation ※Explained with sample code

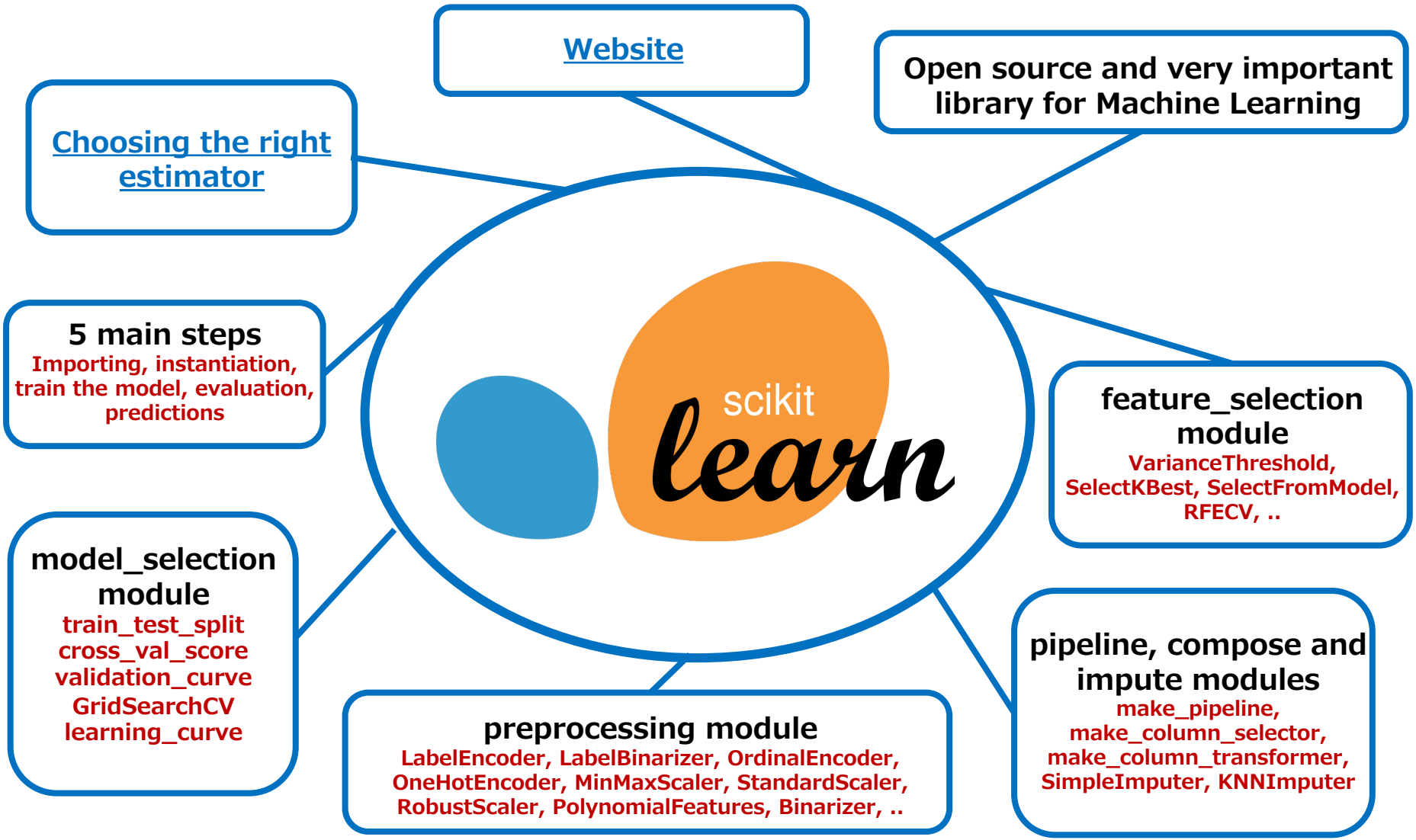
[Problem 2] Grid search ※Explained with sample code

[Problem 3] Survey from Kaggle Notebooks

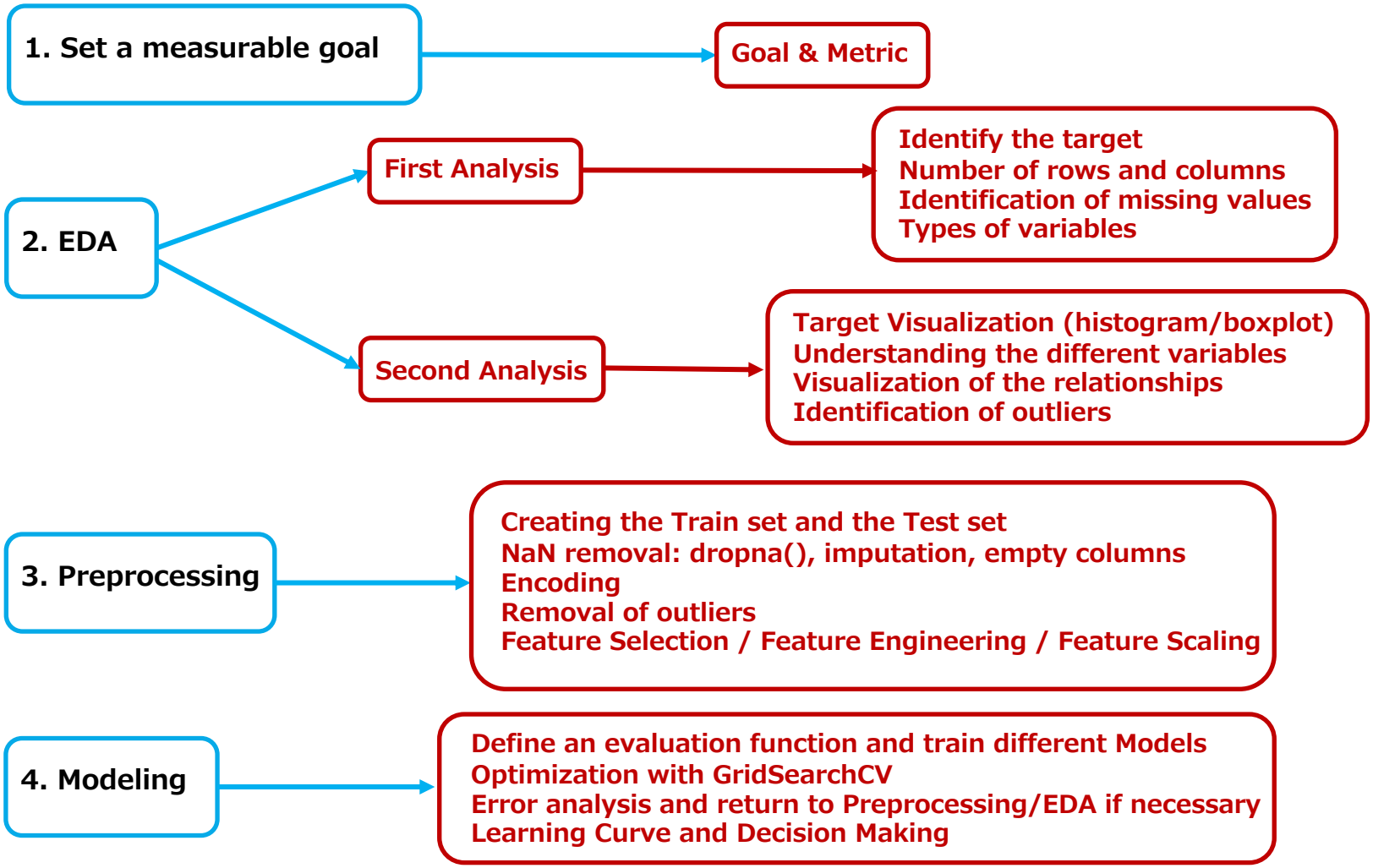
[Problem 4] Creating a model with high generalization performance

[Problem 5] Final model selection

Quick Review (Scikit-learn)



Machine Learning Flow



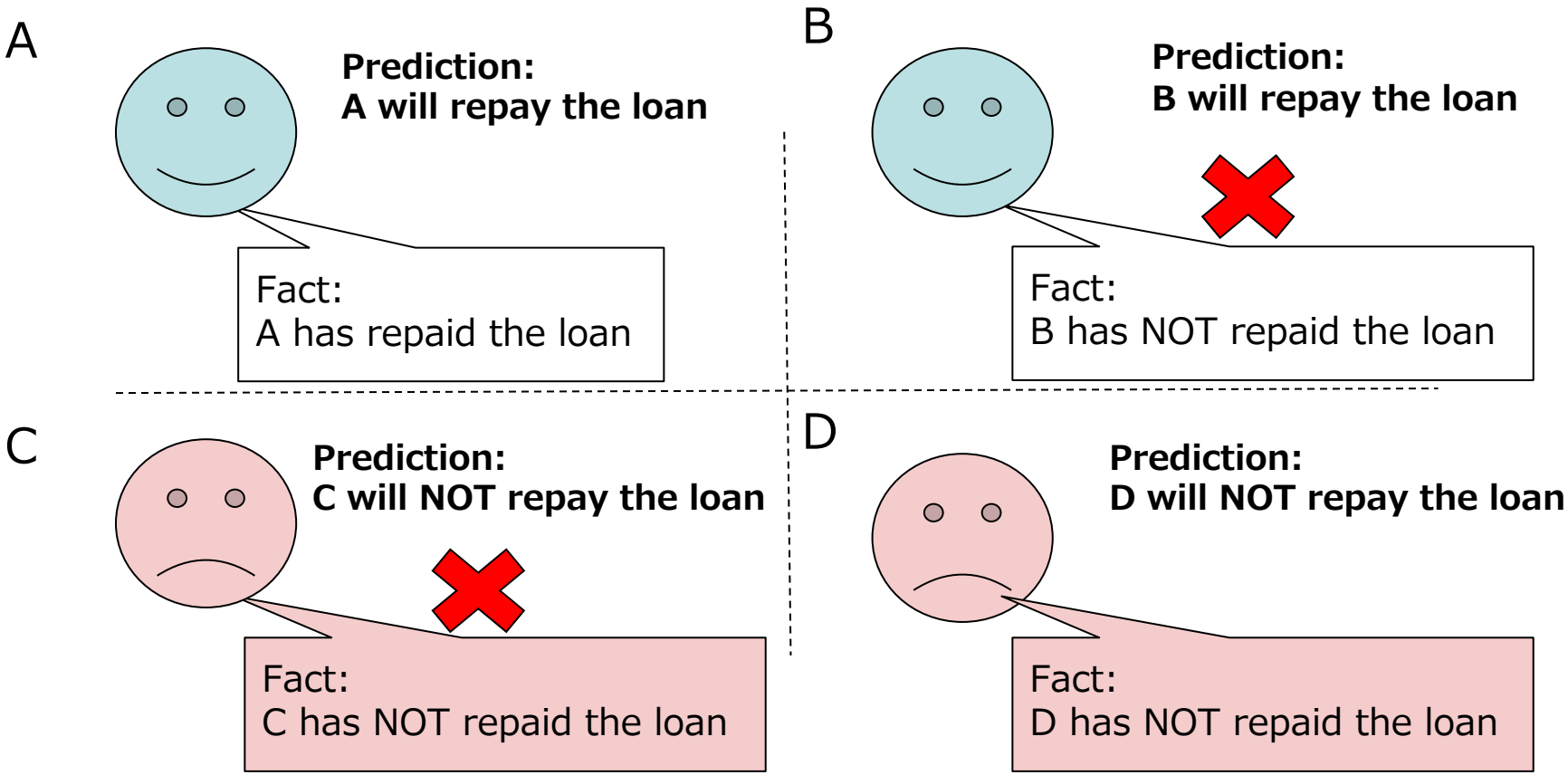
Evaluation of prediction accuracy

Let's consider the accuracy of the classification problem

	Answer	Predicted value	Threshold				
	0 or 1	(probability)	0.1	0.3	0.5	0.7	0.9
1	**1**	0.7	1	1	1	1	0
2	**0**	0.3	1	1	0	0	0
3	**0**	0.8	1	1	1	1	0
4	**0**	0.4	1	1	0	0	0
5	**0**	0.9	1	1	1	1	1
6	**0**	0.5	1	1	1	0	0
7	**0**	0.7	1	1	1	1	0
8	**0**	0.2	1	0	0	0	0
9	**0**	0.5	1	1	1	0	0
10	**0**	0.1	1	0	0	0	0
		percentage of correct answers	10%	30%	50%	70%	80%

Confusion matrix (1/5)

There are other index values for accuracy besides the percentage of correct answers.



Confusion matrix (2/5)

Confusion matrix, a table that allows you to check the accuracy of your predictions against the actual.

Confusion Matrix		Prediction	
		The loan will be repaid (- negative)	**The loan will NOT be repaid (+ positive)**
Fact	**The loan is repaid (- negative)**	**Correct negative**	**NOT correct NOT positive**
	The loan is NOT repaid (+ positive)	**NOT correct NOT negative**	**Correct positive**

Confusion matrix (3/5)

Let's take a look at TN, FN, FP, and TP of the "confusion matrices.

Confusion Matrix		Prediction ※be the subject	
		The loan will be repaid (- negative)	**The loan will NOT be repaid (+ positive)**
Fact	**The loan is repaid (- negative)**	It was really negative. **True Negative (TN)**	It wasn't positive. **False Positive (FP)**
	The loan is NOT repaid (+ positive)	It wasn't negative. **False Negative (FN)**	It was really positive. **True Positive (TP)**

Confusion matrix (4/5)

Let's take note of the frequently used evaluation metrics Precision and Recall.

Indicator	Name	Calculation Formula	Usage perspective
TPR	Fit rate	TP ÷ (TP + FP)	**Percentage of positive predictions that were actually correct.**
FPR	False positive rate	FP ÷ (FP + TN)	**Ratio of Positive to False Positive predictions out of actual Negative predictions.**

TPR		Predict	
		P	**N**
	T	TP	TN
	F	FP	**FN**

FPR		Predict	
		N	**P**
	N	TN	**FP**
	P	FN	TP

Confusion matrix (5/5)

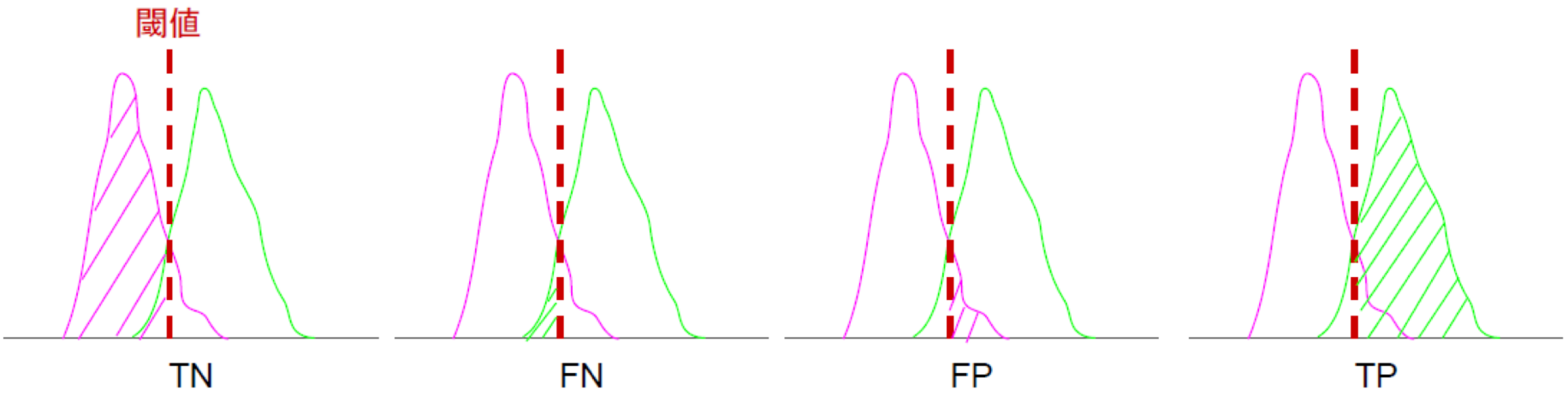
The indicator represented by the confusion matrix can also be represented as a graph of the predictive distribution. Let's have an image of increasing prediction accuracy with thresholds.

		Predict	
		N	P
	N	TN	FP
	P	FN	TP

		Predict	
		N	P
	N	TN	FP
	P	FN	TP

		Predict	
		N	P
	N	TN	FP
	P	FN	TP

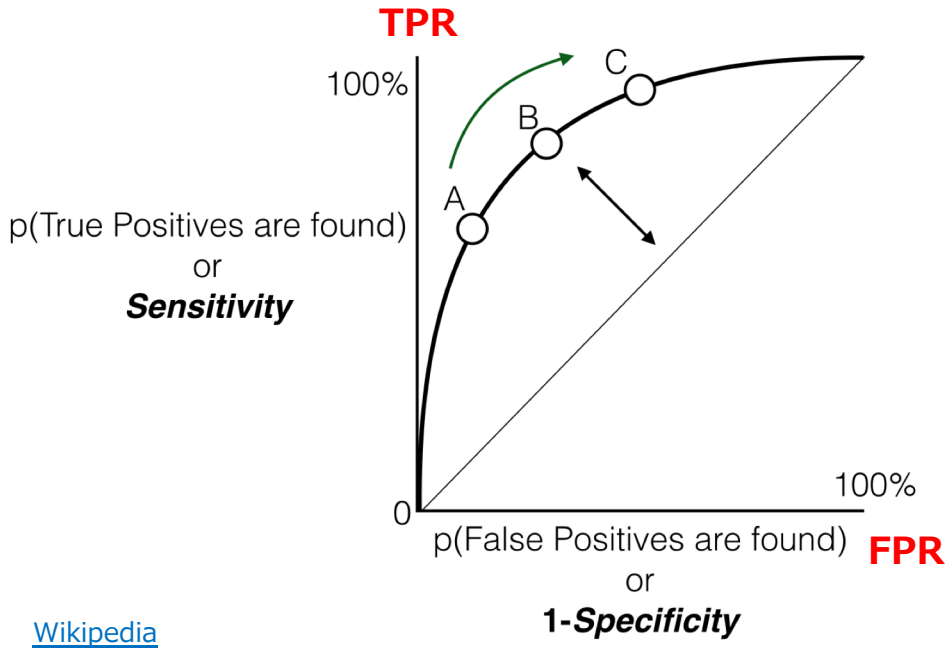
		Predict	
		N	P
	N	TN	FP
	P	FN	TP



Evaluate the model

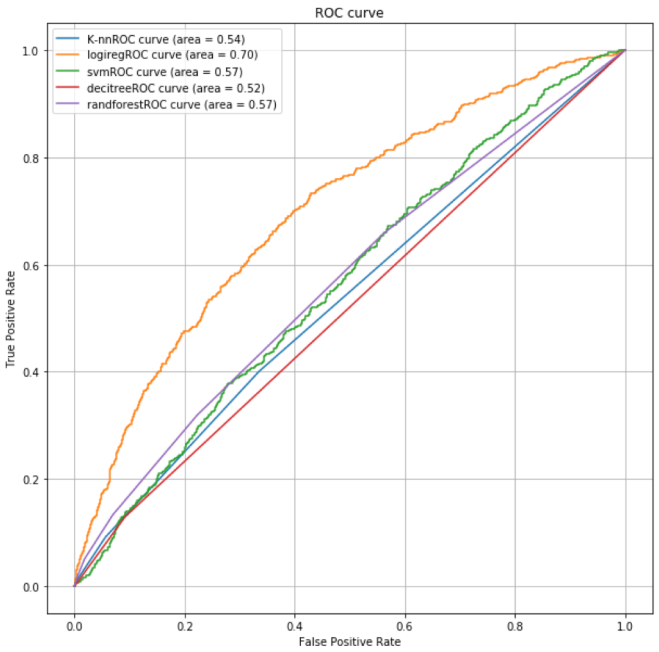
The ROC curve, an evaluation curve with TPR on the vertical axis and FPR on the horizontal axis, is often used to evaluate the performance of a model. This area is the AUC.

Schematic of RoC curve



Wikipedia

Model Performance Evaluation



Cross-validation (1/7)

Kaggle iterative manners

The Form DIC Advisor Yifan Xie's iterative approach https://www.kaggle.com/yifanxie

Many references to cross-validation.

- First create a cross-validation type, then start evaluating the relationship between the cross-validation and the leaderboard score.
- See an experiment to create rigorous cross-validation. This includes cross-parity methodologies, implementations, and data acquisition mechanisms.
- Use cross-validation techniques to determine feature effects and model architecture.

- Understand problem definition
- Understand the evaluation metric in the context of the problem.
- Understand how public and private leaderboard are split
- Initial Data exploration, establish initial understand the nature of dataset
- Build first batch of models, create first batch of submissions
- Create first **cross-validation** scheme, and start evaluating relationship between **cross-validation** and leaderboard score
- Start designing experiment for rigorous **cross-validation**, this shall include both the methodology of **cross-validation**, the actual implementation, a data capturing mechanism for experiment data - In-depth data manipulation for feature engineering,
- For deep learning dominated approaches, design and evaluate model architecture

- use established **cross-validation** method to decide the effectiveness of features and model architecture
- Apply ensemble methods - this can range from simple weigh averaging to stacking
- post-process of prediction from machine learning model. this is optional depends on specific problem and evaluation metric.

The following shall be strongly encouraged throughout the competition:

- Pay close attention to what others are sharing on forum and competition specific kernels
- Review and check past competition top solutions - especially the ones with similar problems and similar evaluation metrics

Cross-validation (2/7)

Why Cross validation ?

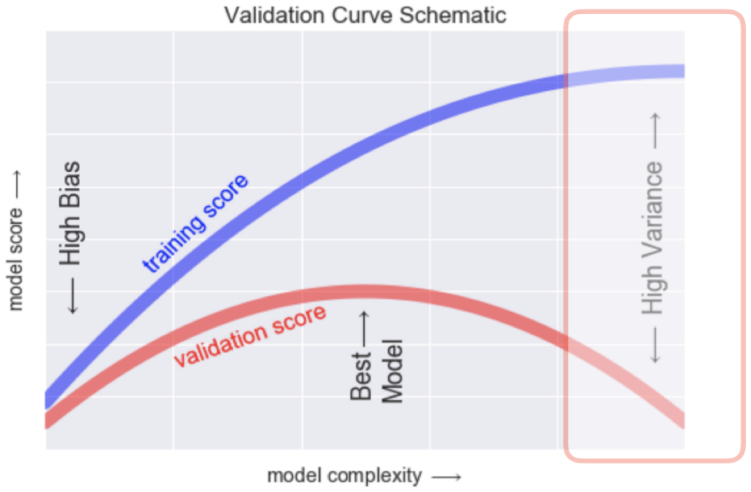
I've got a new feature set
↓
Fit and Pred
↓
Submission
↓
PublicLB's score went up
↓
Done
↓
Wait a second
↓
Could it be a result of overfitting some of the data?

Cross-validation (3/7)

Why Cross validation ?

The goal of machine learning is not to fit the model to the observed data at hand, but to improve the prediction performance for unknown data.
https://jakevdp.github.io/PythonDataScienceHandbook/05.03-hyperparameters-and-model-validation.html

In Kaggle, cross-validation is used to estimate whether Public LB scores are reliable scores or not.
※Since the ranking of Public LB is basically the result of evaluation with a part of the test data, when it is evaluated with the rest of the data (Private LB), the ranking may decrease. In order to obtain stable evaluation results no matter what kind of data is used, cross-validation is used to verify the generalization performance.



Cross-validation (4/7)

What does cross validation do?

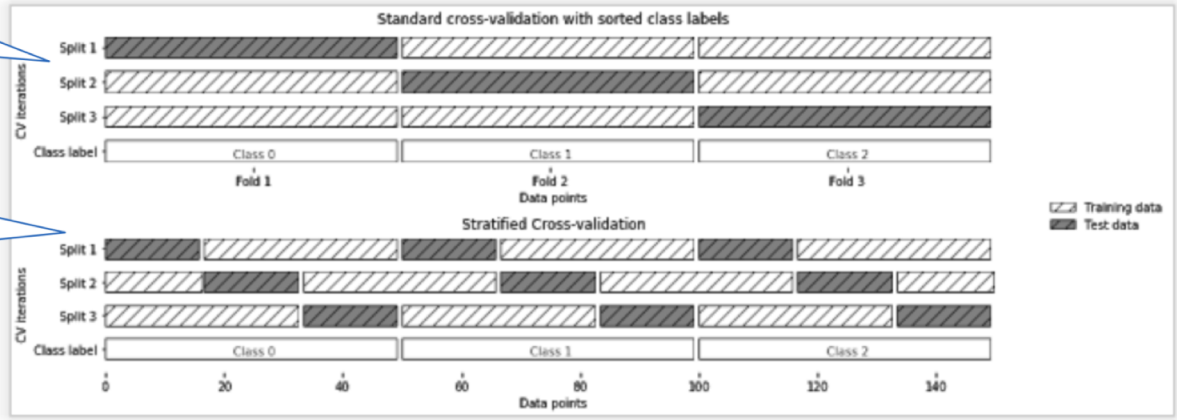
https://machinelearningmastery.com/k-fold-cross-validation/

1. Shuffle the data set randomly.
2. Split the data set into k groups.
3. Perform the following for each split (k times): (4~7)
4. Use one group as a holdout or test dataset
5. Use the rest of the groups as training data set
6. Fit the model to the training set and evaluate it on the test data set
7. Keep the evaluation score and discard the model.
8. Average the model evaluation scores to summarize the model performance

KFold

Stratified KFold

A variation of KFold that returns stratified folds.



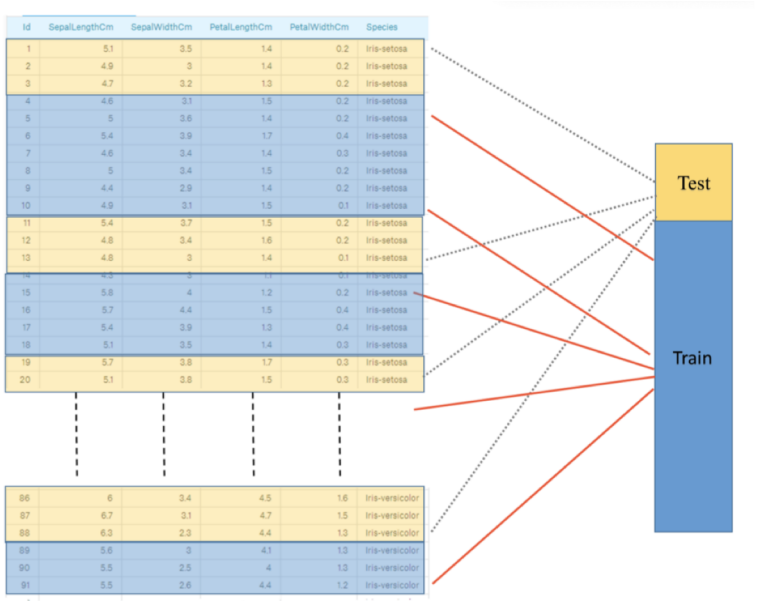
Cross-validation (5/7)

How big you should make K for k-fold cross validation?

K should be which number

https://stats.stackexchange.com/questions/157689/how-big-to-make-k-for-cross-validation

1. **k=1: If the data set is large enough, use the Holdout method (same thing as train_test_split).**
2. **k=4 or 5 or 10: In general, we often see 4, 5, or 10 in analytical competitions.**
3. **k=n: n is the number of records in the data set itself. What to do when the number of records is very small. The approach is called the leave-one-out method.**



Cross-validation (6/7)

Does k-fold cross validation always imply k uniformly sized subsets?

Is the number of divided subsets always equal?

https://stats.stackexchange.com/questions/134266/does-k-fold-cross-validation-always-imply-k-uniformly-sized-subsets

The data is divided as evenly as possible.
When 10-fold cross-validation is specified for 101 datasets, the number of folds is automatically adjusted to 11.

Cross-validation (7/7)

What is good Cross validation?

It depends on how good the validation data can be.
Validation data should be similar to the distribution of test data.

Let's try to set parameters such as shuffle and random_state.

from sklearn.model_selection import KFold, StratifiedKFold

Let's import the sklearn.model_selection class and do a cross validation!
※ StratifiedKFold is a method of dividing a sample into classes while preserving the proportion of each class in the sample.

***Note that sometimes articles use KFold or StratifiedKFold in sklearn.cross_varidation, but that is an old notation.**

http://segafreder.hatenablog.com/entry/2016/10/18/163925

Parameter Tuning (1/4)

How to Tune Algorithm Parameters ?

In sklearn models, hyperparameters can be specified, which can be tuned to improve the accuracy of the model. So, what tuning methods are available?

Grid Search

Systematically build and evaluate the model for each combination of algorithm parameters specified in the grid (Grid Search is the most time-consuming.)

Random Search

Sample parameters from a random distribution (i.e., a uniform distribution representing random events) for a fixed number of iterations.
Sampling parameters from a random distribution (i.e., a uniform distribution, which is a probability distribution that represents random events) for a fixed number of iterations

Bayesian Optimization(Rederence): https://github.com/fmfn/BayesianOptimization

Parameter Tuning (2/4)

What does Grid Search do?

Create a grid of possible values of the hyperparameters and exhaustively find a good combination of model evaluations (among all the grid points)

Example:SVM

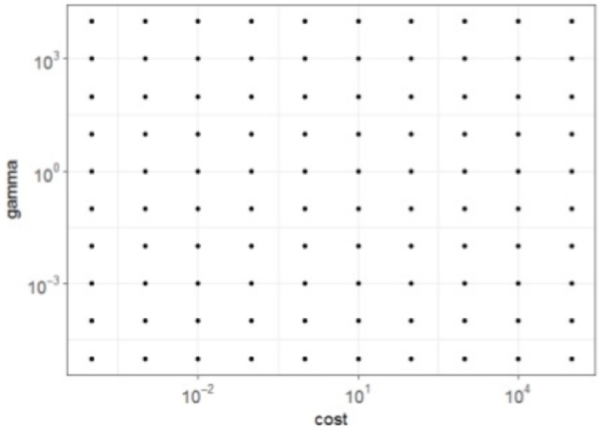
Gamma = (10^-5, 10^-4, … , 10^3, 10^4)

 # The smaller the size, the simpler the decision boundary.

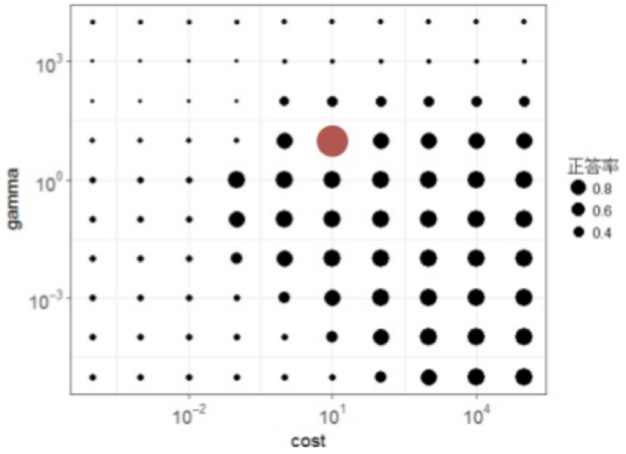
Cost = (10^-4, 10^-3, … , 10^4, 10^5)

 # The smaller the number, the more acceptable the misclassification. The smaller it is, the more misclassification is allowed.

Make the search range grid-like



Find the correct answer rate at all grid points



Parameter Tuning (3/4)

In Grid Search (Sklearn), what metrics are used to evaluate the search results?

Regression Problems: Coefficient of Determination (R^2)
Classification problems: Percentage of correct predictions

Parameter Tuning (4/4)

Model	Parameters to optimize	Good range of values
Linear Regression	• fit_intercept • normalize	• True / False • True / False
Ridge	• alpha • Fit_intercept • Normalize	• 0.01, 0.1, 1.0, 10, 100 • True/False • True/False
k-neighbors	• N_neighbors • p	• 2, 4, 8, 16 …. • 2, 3
SVM	• C • Gamma • class_weight	• 0.001, 0.01…..10...100...1000 • 'Auto', RS* • 'Balanced' , None
Logistic Regression	• Penalty • C	• L1 or l2 • 0.001, 0.01…..10...100
Naive Bayes (all variations)	NONE	NONE
Lasso	• Alpha • Normalize	• 0.1, 1.0, 10 • True/False
Random Forest	• N_estimators • Max_depth • Min_samples_split • Min_samples_leaf • Max features	• 120, 300, 500, 800, 1200 • 5, 8, 15, 25, 30, None • 1, 2, 5, 10, 15, 100 • 1, 2, 5, 10 • Log2, sqrt, None
Xgboost	• Eta • Gamma • Max_depth • Min_child_weight • Subsample • Colsample_bytree • Lambda • alpha	• 0.01,0.015, 0.025, 0.05, 0.1 • 0.05-0.1,0.3,0.5,0.7,0.9,1.0 • 3, 5, 7, 9, 12, 15, 17, 25 • 1, 3, 5, 7 • 0.6, 0.7, 0.8, 0.9, 1.0 • 0.6, 0.7, 0.8, 0.9, 1.0 • 0.01-0.1, 1.0 , RS* • 0, 0.1, 0.5, 1.0 RS*

Examples of hyperparameter search ranges
This is one example from the Kaggle blog. The optimal parameter search range changes depending on the dataset, so this may not always work.
Also, if there are too many hyperparameters to be explored or if the search area is expanded too much, it will take a huge amount of time, so start with a small number of options.

Notation examples for scikit-learn:

```
clf =
GridSearchCV(estimator=RandomForestClassifier(),
 param_grid=parameter_candidates,
                    cv=5,
                    refit=True,
                    error_score=0,
                    n_jobs=-1)
clf.fit(data_X, data_y)
cv_result = pd.DataFrame(clf.cv_results_)
 cv_result # cv_results_ attribute to see the training results for each parameter
```

https://scikit-learn.org/stable/modules/grid_search.html

Kaggle submission

Once the validation (cross-validation) is done, we train on that model with all the training data and make predictions for the test data.

The model instance used for training in validation is not used for this last model for estimation (as it is standard).

Once you have a prediction, look here to create a Submission File.

https://www.kaggle.com/dansbecker/submitting-from-a-kernel

↑Let's learn how to prepare a Submission File from the kernel of House Price here. Note that the column names are for House Price!

Sprint 1 – Machine Learning Flow

Explanation about this Sprint will be given in the mentoring sessions but please try it on your own first.

Sprint 1 – Machine Learning Flow

Please work on your own after class and submit your assignments on DIVER.

Sprint 1 – Sample Code

A Sample Code of this Sprint is given but please try it on your own.

Sprint 1 – Machine Learning Flow

Please work on your own after class and submit your assignments on DIVER.

ToDo by next class

Next class will be held on Zoom : Thursday 6 May 2021

🚨ToDo: Sprint 2 – Introduction to Machine Learning Scratch
https://diver.diveintocode.jp/curriculums/1643

Check-out

3 minutes Please post the following point to Zoom chat.

Q. Current feelings and reflections
(joy, anger, sorrow, anticipation, nervousness, etc.)

Thank You For Your Attention