

Machine Learning for Finance (FIN 570)

Hyperparameter Tuning: Bias-Variance Tradeoff, Cross-Validation, and Evaluation Metric

Instructor: Jaehyuk Choi

Peking University HSBC Business School, Shenzhen, China

2018-19 Module 1 (Fall 2018)

Measuring quality of ML method

Given a ML method, we want to minimize the mean squared error (MSE) on **test data set** (expected test MSE).

$$E\left(y - \hat{f}(x)\right)^2 = \text{Bias}(\hat{f}(x))^2 + \text{Var}(\hat{f}(x)) + \text{Var}(\varepsilon)$$

where, $y = f(x) + \varepsilon$ (true pattern)

- By *a given ML method*, we mean that model (LR, SVM, etc) and hyper-parameter (C , γ , reduced dimension k for PCA/LDA, etc) are fixed. However fitted model parameters (i.e., \hat{f}) can change over training set.
- The expectation is made over repeatedly selecting different training vs test dataset. Therefore, the expectation is over \hat{f} as well as x .
- We need to minimize $\text{Bias}(\hat{f}(x))^2$ and $\text{Var}(\hat{f}(x))$ together while $\text{Var}(\varepsilon)$ is fundamentally irreducible.

Bias and Variance

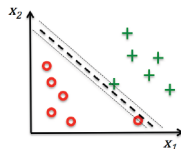
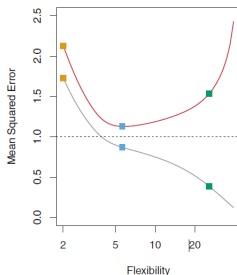
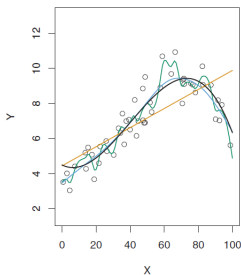
Bias

- Error from \hat{f} not correctly representing the true f (e.g. linear regression on non-linear data).
- A model has **high bias** when \hat{f} overly simply f (under-fitting), i.e., the used parameters are too few.

Variance

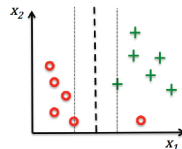
- Error from variability or sensitivity (vs consistency) of the trained model \hat{f} against the selection of training dataset.
- A model has **high variance** when the model is too flexible (over-fitting), i.e., there are too many parameters, e.g. KNN with $K = 1$, high-order polynomial regression, SVM/LR with large C (small λ), decision tree with many leaves, etc.

Bias and Variance (examples)



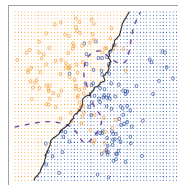
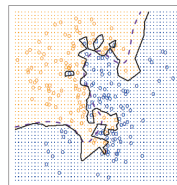
Large value for parameter C

KNN: $K=1$

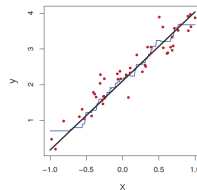
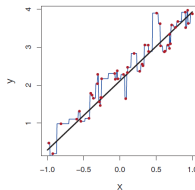


Small value for parameter C

KNN: $K=100$

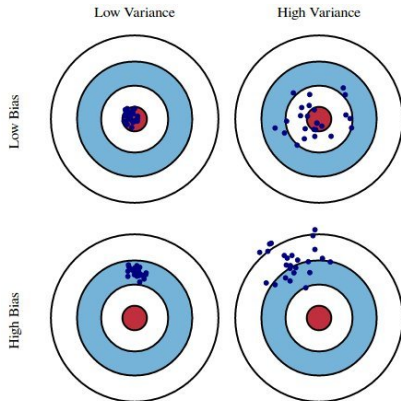


- Grey line: Bias vs the number of parameters
- Red line: MSE measured with the true f (black line).



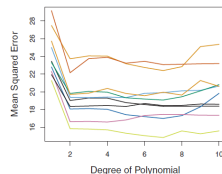
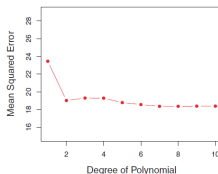
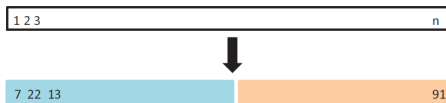
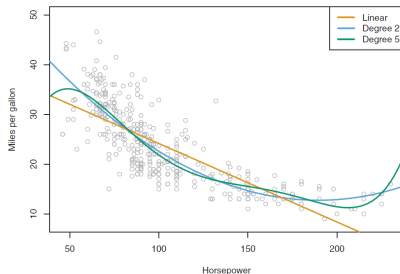
Bias-Variance Tradeoff

- It is hard to reduce both bias and variance.
- As the model flexibility increase, bias decreases but variance increases. It is important to find a right trade-off.
- Bias-variance tradeoff is one of the most important theme in ML (and other fields!).
- In real problems, the true pattern f is unknown and the dataset size is limited. How can we efficiently measure the expected test MSE?



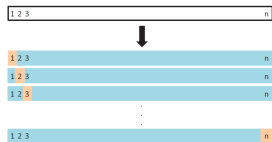
Cross-Validation(CV): Validation Set (Hold-out set)

- Divide observations into a **training set** and a **validation (hold-out) set**.
- Fit model on the training set and measure error on the validation set.
- Error rate is highly variable (sensitive to division) and over-estimated than the true test error rate as the model is trained on fewer observations.
- Training set is further divided into **training** and **validation** sets. Validation set is used for model selection and hyper-parameter tuning.

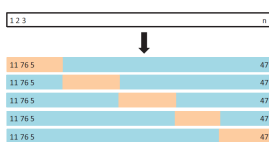
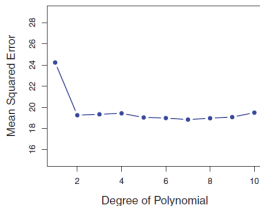


Cross-Validation: Leave-One-Out (LOOCV) and k -Fold CV

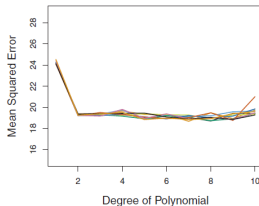
- LOOCV: train model with one sample left out and measure the error on the sample. Error is close to the true test rate but computation is heavy (train n times).
- k -fold CV: divide the samples into k (typically 5 or 10) folds. Train model on $k - 1$ **training** folds and measure error on the remaining **test** fold.



LOOCV



10-fold CV



Evaluation Metrics

Confusion Matrix

Credit card Default		Predicted		
		P^*	N^*	Total
Actual	P	40	40	80
	N	10	910	920
	Total	50	950	1000

		Predicted class	
		P^*	N^*
Actual class	P	True positives (TP)	False negatives (FN)
	N	False positives (FP)	True negatives (TN)

- Accuracy (ACC) = $\frac{TP + TN}{ALL} = \frac{40 + 910}{1000} = 95\%$
- Error (ERR) = $1 - ACC = \frac{FP + FN}{ALL} = \frac{10 + 40}{1000} = 5\%$
- However, accuracy/error may be misleading!

Evaluation Metrics

		Predicted	
		P^*	N^*
Actual	P	TP (40)	FN (40)
	N	FP (10)	TN (910)

- Precision (PRE) = $\frac{TP}{P^*} = \frac{TP}{TP + FP} = \frac{40}{50} = 80\%$

Case: Spam mail filter (minimize FP)

- Recall (REC) = $\frac{TP}{P} = \frac{TP}{TP + FN} = \frac{40}{80} = 50\%$

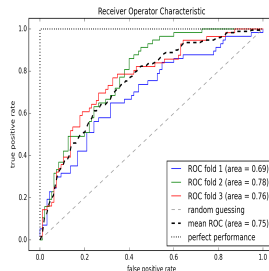
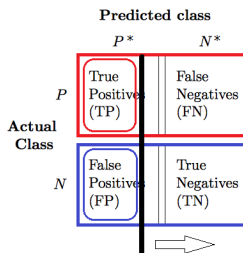
Case: Credit approval, Cancer diagnosis (minimize FN)

- F1-Score (F1) = $\frac{2 \text{ PRE} \times \text{REC}}{\text{PRE} + \text{REC}} = 61.5\% \quad \left(\frac{2}{F1} = \frac{1}{\text{PRE}} + \frac{1}{\text{REC}} \right)$

The harmonic average of PRE and REC to ensure $0 \leq F1 \leq 1$

A widely used accuracy for binary classification with imbalanced sample.

Receiver Operator Characteristic (ROC) Curve



- True Positive Rate ($\text{TPR}=\text{REC}$) = $\text{TP}/P = 50\%$
- False Positive Rate (FPR) = $\text{FP}/N = 10/920 = 1.1\%$
- ROC Curve: graph of (FPR, TPR) for varying classification threshold of the binary classification.
- Area Under Curve (AUC) give an overall accuracy of a classifier, summarizing over all possible threshold
- The diagonal line is from random-guessing: ROC AUC = 0.5
A model with lower AUC than 0.5 is worthless.
- A perfect classifier (I-shaped lines): ROC AUC = 1.