Model Assessment and Selection

Yuanning Li

BME 2111
Neural Signal Processing and Data Analysis
2023 Fall

Model assessment and selection

- We have already covered many models
  - Classifications: naive Bayes, probabilistic generative model, LDA, logistic regression, SVM
  - Clustering: K-means, Gaussian mixtures
  - Dimensionality reduction: PCA, PPCA, FA

- How do we quantify the performance of these models?
- Do they have hyper-parameters? How do we choose between different models and/or different hyper-parameters?

Model assessment and selection

- For a given data set, there may be several candidate models. How do we choose among them?
- For a given model, how do we choose the appropriate level of complexity?
  - e.g. For the Gaussian mixture model, how do we choose the number of clusters K?

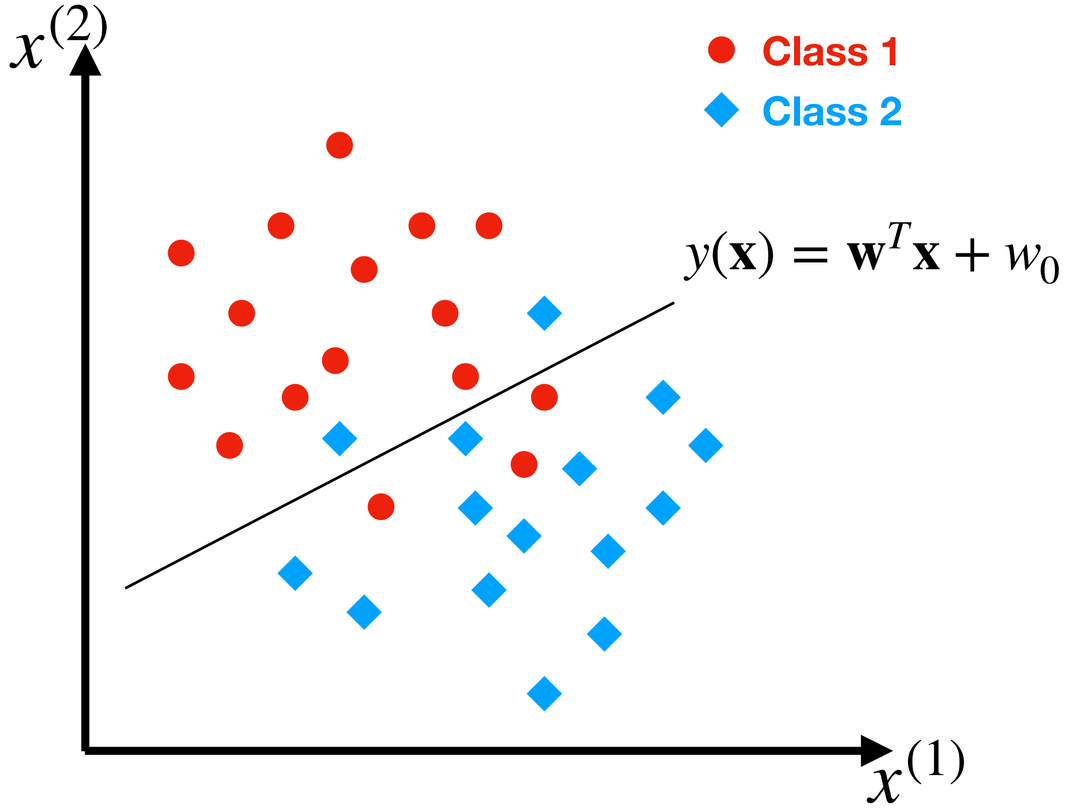
Model assessment and selection

- One way to answer these questions is to assess generalization performance.
- We want to choose the model and level of complexity that has the greatest predictive ability on test data (which was not used to train the model).

Quantification of Performance: Prediction error

Examples:

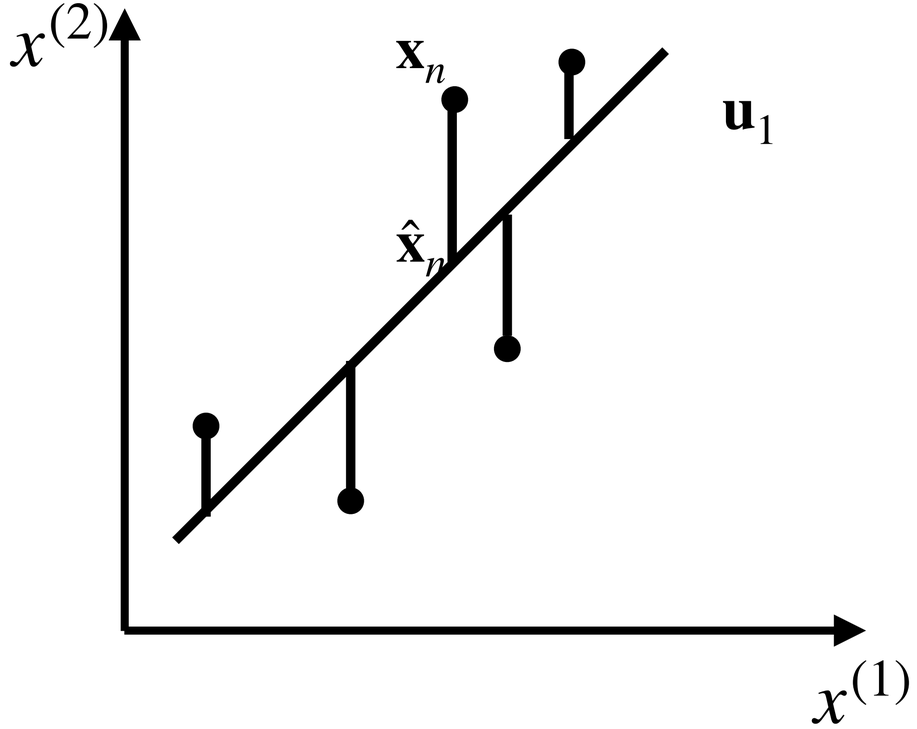
- Classification
  - What is the % of data points incorrectly classified?



Quantification of Performance: Prediction error

Examples:

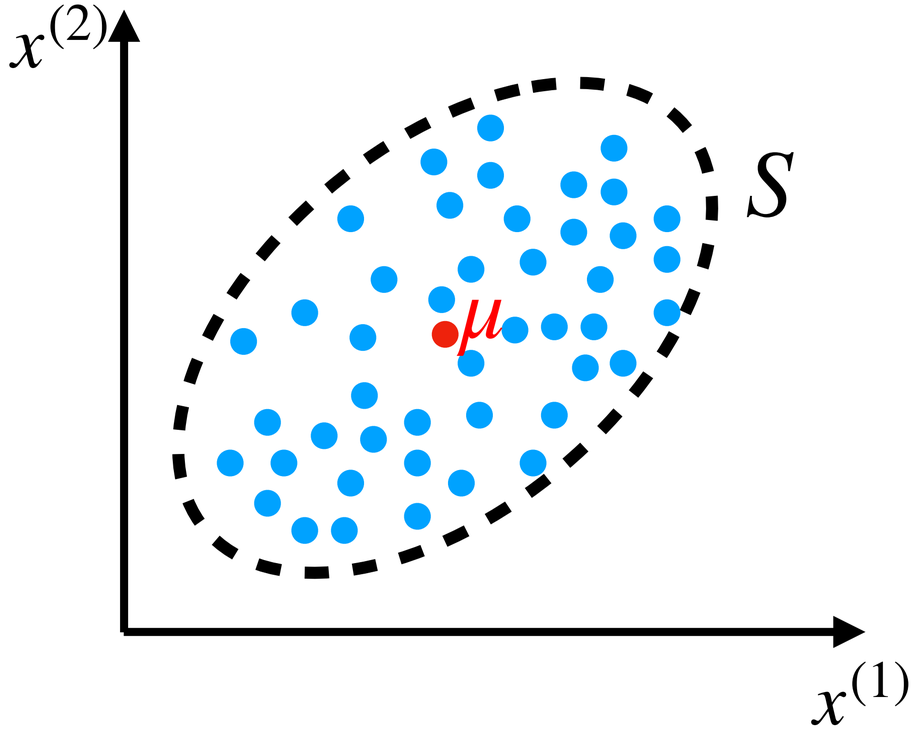
- Regression
  - What is the sum of squared errors?



Quantification of Performance: Data likelihood

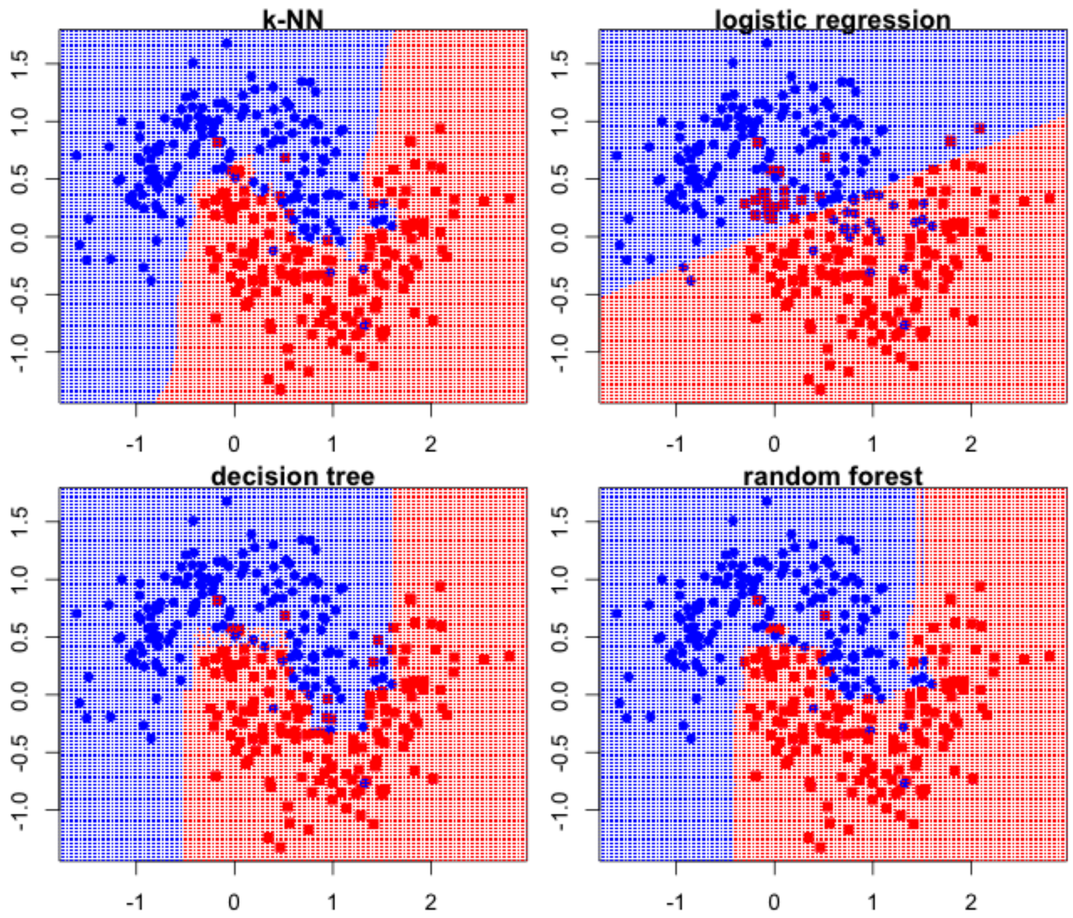
Examples:

- What is the probability of the data under the model $P(X \mid \theta)$
- In some cases, the prediction error can be interpreted as a data likelihood.



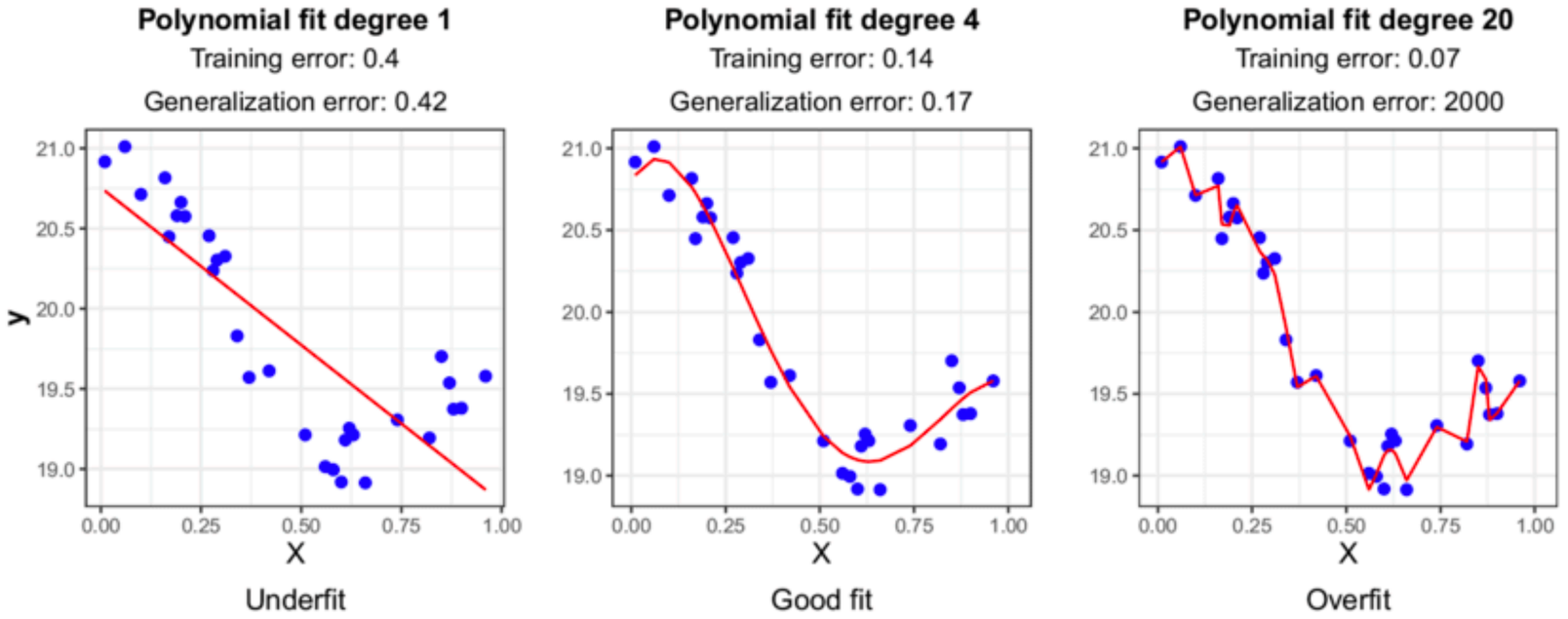
Model Complexity

- Classification: How wiggly is the decision boundary?



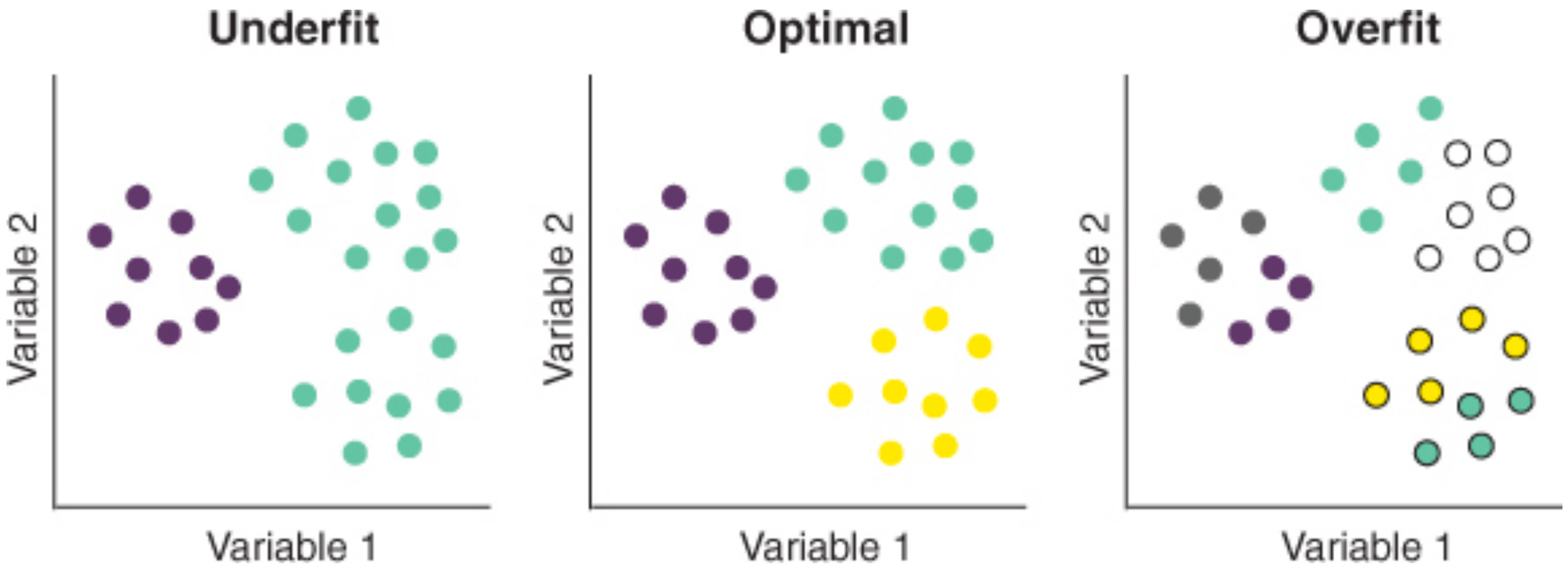
Model Complexity

- Regression: How wiggly is the regression line?



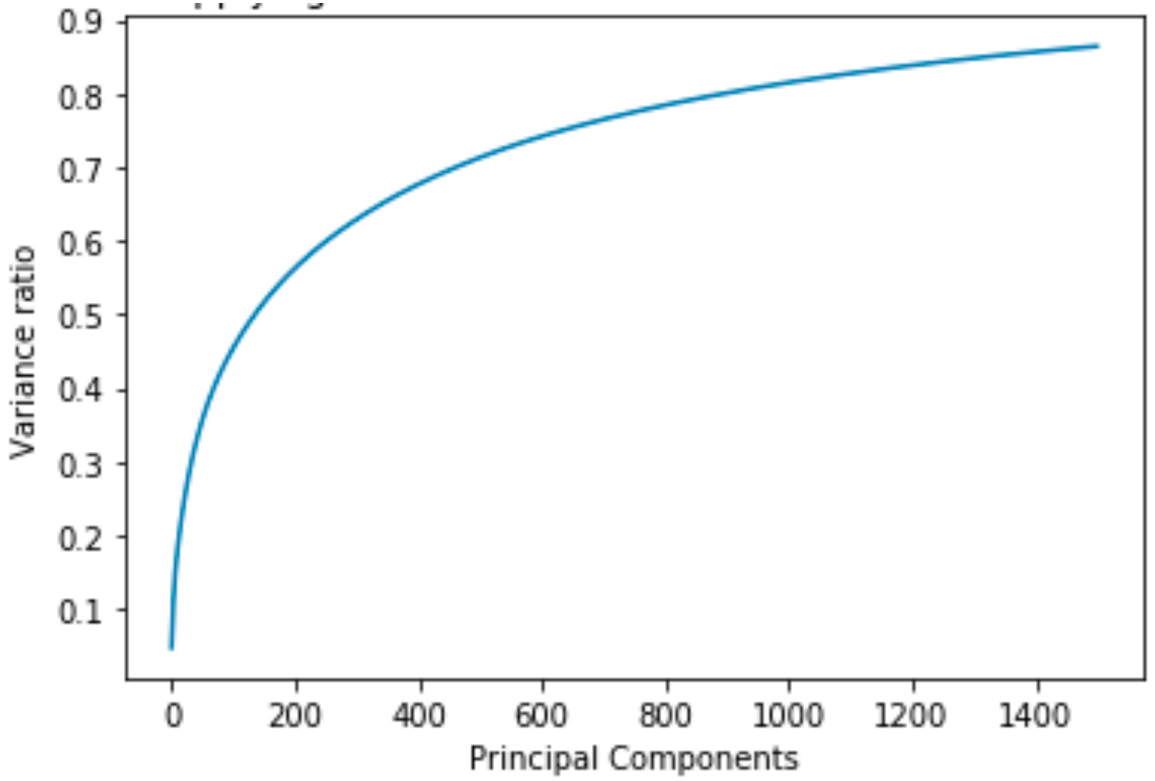
Model Complexity

- Clustering: How many mixture components?



Model Complexity

- Dimensionality reduction: How many latent variables?



What you should know about modeling

- Statistical prediction
  - Suppose that we observe (X, Y) from some unknown joint distribution, and we want to predict Y from X. Over all functions f, the expected prediction error, measured in terms of squared loss $\mathbb{E}\left[(Y - f(X))^2\right]$ is minimized at $f(x) = \mathbb{E}(Y \mid X = x)$. This is called the true regression function associated with the pair (X, Y).
  - We can always write $Y = f(X) + \epsilon$, with noise term ϵ satisfying $\mathbb{E}[\epsilon] = 0$
  - However, $f(x) = \mathbb{E}(Y \mid X = x)$ is optimal but unknown in practice.

Statistical prediction

- Now suppose we observe training samples (x_1, y_1), …, (x_n, y_n) i.i.d. from the same joint distribution as (X, Y). We use these training samples to construct a prediction function $\hat{f}$. The expected prediction error, or expected test error, of $\hat{f}$ is

$$\mathbb{E}\left[\left(Y - \hat{f}(X)\right)^2\right] \qquad (1)$$

  where the expectation is over all that is random, namely the training set $\{(x_i, y_i) \mid i = 1,\ldots,n\}$ and the test point (X, Y)

- Why would we be interested in (1)? Here are two reasons:
  - Model assessment: we want to know how well we can predict a future observation, in absolute terms
  - Model selection: we want to choose between different models (e.g. two different model classes, or choosing tuning hyperparameter for a particular method)

Bias-variance tradeoff

- The expected test error in (1) has an important property: it decomposes into informative parts. But before we show this, let's think about a few points:
  - Can we ever predict Y from X with zero error? Generally, no. Even the true regression function f cannot generically do this, and will incur some error due to noise. We call this *irreducible error*.
  - What happens if our fitted function $\hat{f}$ belongs to a model class that is far from the true regression function f? E.g., we choose to fit a linear model in a setting where the true relationship is far from linear? As a result, we encounter error, what we call *estimation bias*.
  - What happens if our fitted (random) function $\hat{f}$ is itself quite variable? In other words, over different copies of the training set, we end up constructing substantially different functions $\hat{f}$? This is another source of error, that we'll call *estimation variance*.

Bias-variance tradeoff

- Formally speaking, for model $Y = f(X) + \epsilon$, where ϵ has mean 0, variance σ^2, and is independent of X. It follows that

$$\mathbb{E}[(Y - \hat{f}(x))^2 \,|\, X = x] = \sigma^2 + \underbrace{\mathbb{E}\left[\left(f(x) - \hat{f}(x)\right)^2\right]}_{\text{Risk}(\hat{f}(x))}$$

- The first term σ^2 is called the *irreducible error*, or sometimes referred to as the Bayes error, and the second term is called the risk, or mean squared error (MSE). The risk further decomposes into two parts

$$\mathbb{E}[(Y - \hat{f}(x))^2 \,|\, X = x] = \sigma^2 + \underbrace{\left(f(x) - \mathbb{E}\left[\hat{f}(x)\right]\right)^2}_{\text{Bias}^2(\hat{f}(x))} + \underbrace{\mathbb{E}\left[\left(\hat{f}(x) - \mathbb{E}[\hat{f}(x)]\right)^2\right]}_{\text{Var}(\hat{f}(x))} \qquad (2)$$

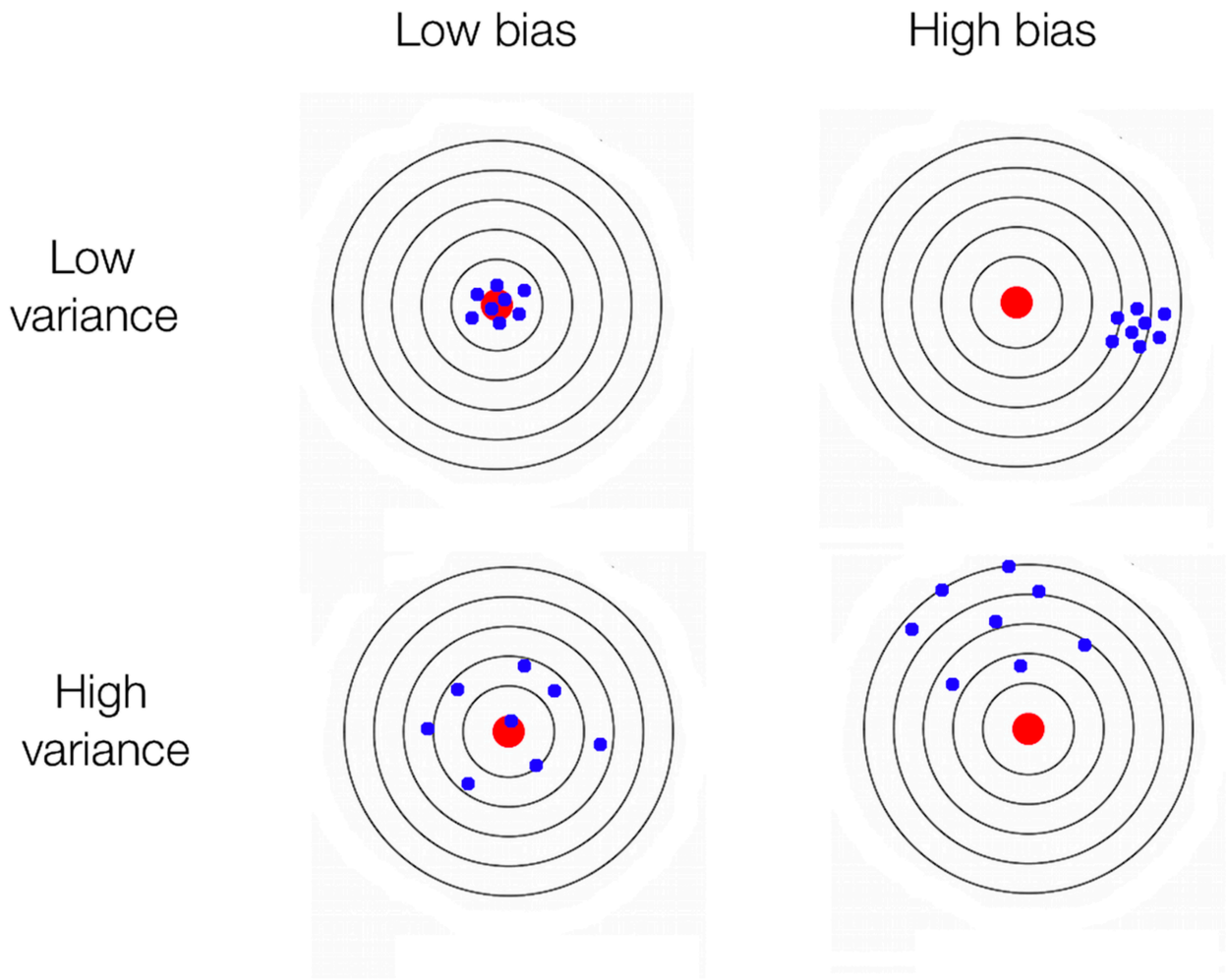
  The latter terms being the squared *estimation bias* or simply *bias*, and the *estimation variance* or simply *variance*.

- The decomposition (2) is called the *bias-variance decomposition* or *bias-variance tradeoff*.

- Integration (2) over X gives $\mathbb{E}\left[\left(Y - \hat{f}(X)\right)^2\right] = \sigma^2 + \int \text{Bias}^2(\hat{f}(x))P_X(dx) + \int \text{Var}(\hat{f}(x))P_x(dx)$

  expected test error = Bayes error + average bias + average variance.

Bias-variance tradeoff



Estimating the error term

- How to estimate the expected test error in (1)? If we had an independent test set $\{(x_i', y_i') \mid i = 1,\ldots,m\}$, the the observed average test error $\frac{1}{m}\sum_{i=1}^{m}\left(y_i' - \hat{f}(x_i')\right)^2$ would serve as an unbiased estimate for $\mathbb{E}[(Y - \hat{f}(x))^2]$. But we are often not this fortunate to have enough test data.

Estimating the training error

- What's wrong with the average training error $\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$?

- This measures the squared error of $\hat{f}$ around the data we used to fit it. The problem is that this would be far too optimistic.
- The expected test and training errors can be linked in a useful way, if we assume that fixed inputs, for simplicity. That is, let $(x_1, y_1'), \ldots, (x_n, y_n')$ denote a test set, independent of and having the same distribution as $(x_1, y_1), \ldots, (x_n, y_n)$. The inputs $x_1, \ldots, x_n$ are fixed in both sets, and only the observations y_i, y_i', are random, drawn according to

$$y_i = f(x_i) + \epsilon_i, \qquad i = 1,\ldots n$$
$$y_i' = f(x_i) + \epsilon_i', \qquad i = 1,\ldots n$$

all errors ϵ_i, ϵ_i', $i = 1,\ldots n$ i.i.d. with mean 0 and variance σ^2.

Estimating the training error: optimism

- Then the expected test error is $\frac{1}{n}\mathbb{E}\|y' - \hat{f}\|_2^2$ where the expectation is taken over y, y'. Now write $f = \big(f(x_1), \ldots, f(x_n)\big) \in \mathbb{R}^n$, and decompose

$$\begin{aligned}\frac{1}{n}\mathbb{E}\|y' - \hat{f}\|_2^2 &= \frac{1}{n}\mathbb{E}\|y' - f + f - \hat{f}\| = \sigma^2 + \frac{1}{n}\mathbb{E}\|f - \hat{f}\|_2^2 \\ &= 2\sigma^2 + \frac{1}{n}\mathbb{E}\|y - \hat{f}\|_2^2 + \frac{2}{n}\mathbb{E}(f - y)^T(y - \hat{f}) \\ &= \frac{1}{n}\mathbb{E}\|y - \hat{f}\|_2^2 + \frac{2}{n}\text{tr}\left(\text{Cov}(y, \hat{f})\right)\end{aligned}$$

- The second term is the ***optimism*** — difference in expected test and expected training errors. The higher the correlation between y_i and its fitted value $\hat{f}(x_i)$, the greater the optimism.

Degrees of freedom

- For $\hat{f} = \left(\hat{f}(x_1), \ldots, \hat{f}(x_n)\right) \in \mathbb{R}^n$ and the same setup as before, the degrees of freedom of $\hat{f}$ is defined as

$$\text{df}(\hat{f}) = \frac{1}{\sigma^2} \sum_{i=1}^{n} \text{Cov}\left(y_i, \hat{f}(x_i)\right)$$

- Hence the optimism can be written as

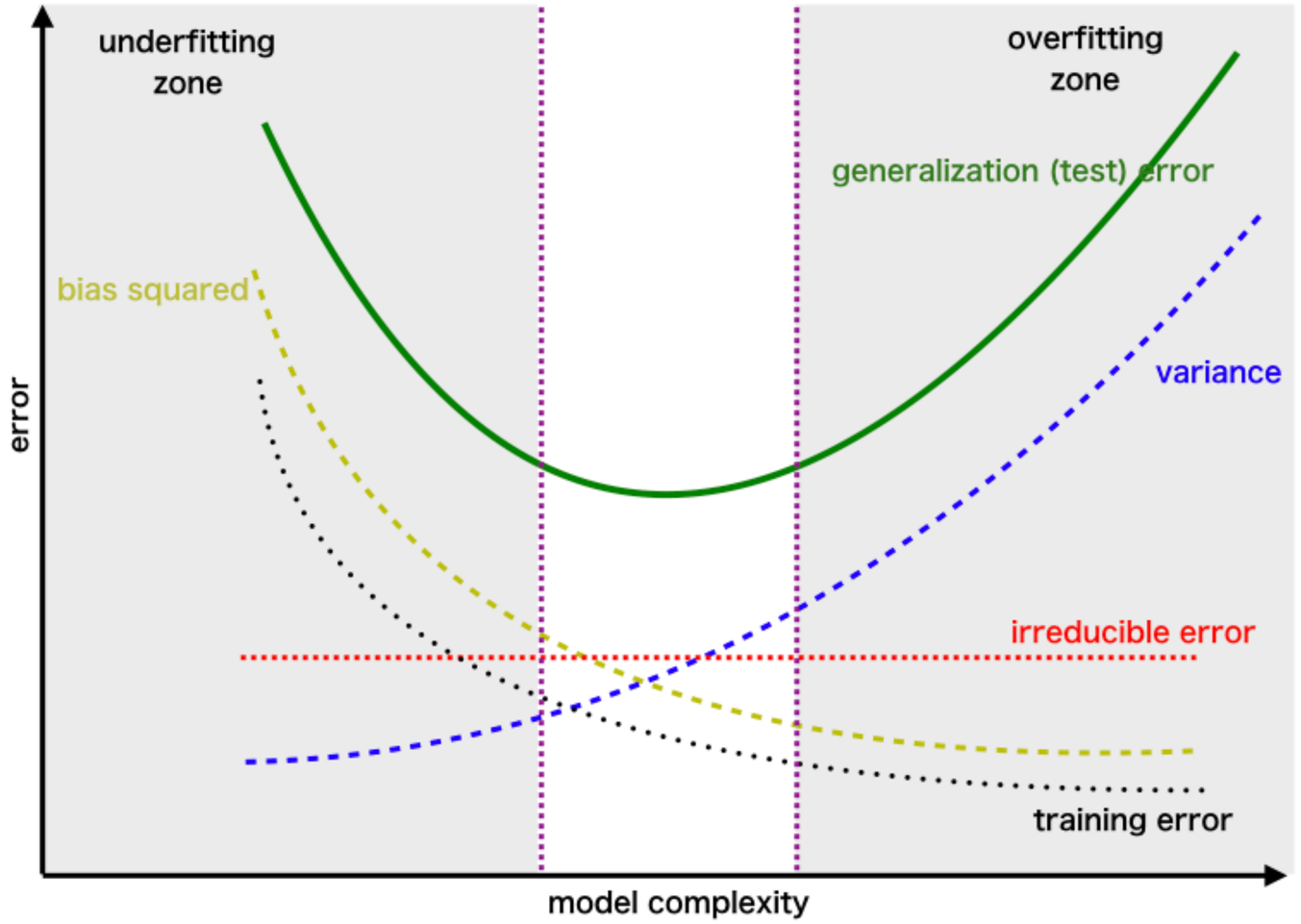
$$\frac{1}{n}\mathbb{E}\|y' - \hat{f}\|_2^2 - \frac{1}{n}\mathbb{E}\|y - \hat{f}\| = \frac{2\sigma^2}{n}\text{df}(\hat{f})$$

- Intuitively, we can think of $df(\hat{f})$ as the effective number of parameters used by the fit $\hat{f}$, a measure of complexity of $\hat{f}$. A few simple examples that support this intuition:
  - If $\hat{f} = (y_1, \ldots, y_n)$, then $\text{df}(\hat{f}) = n$
  - If $\hat{f}(x_i) = \bar{y} = \frac{1}{n}\sum_{j=1}^{n} y_j$ for each $i = 1,\ldots,n$, then $\text{df}(\hat{f}) = 1$
  - If $\hat{f}(x_i) = \frac{1}{k}\sum_{j \in \mathcal{N}_k(x)} y_j$ for each $i = 1,\ldots,n$, then $\text{df}(\hat{f}) = k$

Degrees of freedom

- Suppose we can compute the degrees of freedom of $\hat{f}$, or even more broadly, compute an unbiased estimate $\widehat{\text{df}}\,(\hat{f})$, with $\mathbb{E}\left[\widehat{\text{df}}\,(\hat{f})\right] = \text{df}(\hat{f})$
- Then the quantity $\hat{T} = \frac{1}{n}\|y - \hat{f}\|_2^2 + \frac{2\sigma^2}{n}\widehat{\text{df}}\,(\hat{f})$ serves as unbiased estimate for the expected test error of $\hat{f}$, i.e. $\mathbb{E}(\hat{T}) = \frac{1}{n}\mathbb{E}\|y' - \hat{f}\|_2^2$
- Unbiased risk estimation $\hat{R} = \hat{T} - \sigma^2 = \frac{1}{n}\|y - \hat{f}\|_2^2 + \frac{2\sigma^2}{n}\widehat{\text{df}}\,(\hat{f}) - \sigma^2$ with $\mathbb{E}(\hat{R}) = \frac{1}{n}\mathbb{E}\|f - \hat{f}\|_2^2$

Picture to have in mind

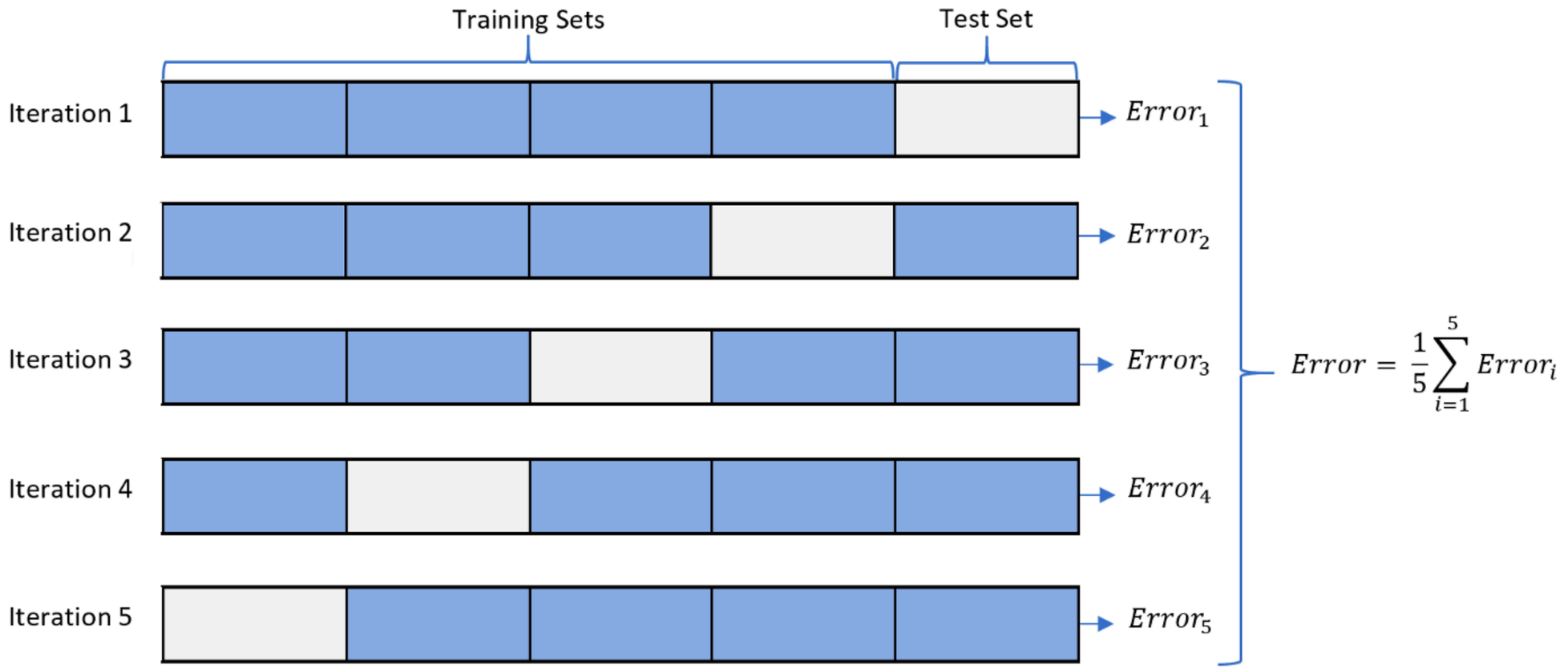


Model selection

- An example of how to use this for model selection: suppose that we were considering fitted models $\hat{f}$ indexed by a parameter $\theta \in \Theta$;
- Suppose also that had access to $\widehat{df}(\hat{f}_\theta)$ at each value of θ. Then we could choose the parameter by minimizing our test error estimate,

$$\hat{\theta} = \underset{\theta \in \Theta}{\text{argmin}} \frac{1}{n}\|y - \hat{f}_\theta\|_2^2 + \frac{2\sigma^2}{n}\widehat{\text{df}}(\hat{f}_\theta)$$

Cross-validation



Cross-validation

- Cross-validation (CV) is quite a general tool for estimating the expected test error (1), that makes minimal assumptions:
  - it doesn't assume that $Y = f(X) + \epsilon$ with ϵ independent of X,
  - it doesn't assume that the training inputs $x_1, \ldots, x_n$ are fixed,
  - all it really assumes is that the training samples $(x_1, y_1), \ldots, (x_n, y_n)$ are i.i.d.

Cross-validation

- We split up our training set into K divisions or folds, for some number K; usually this is done randomly.
- Write these as $F_1, \ldots, F_K$, so $F_1 \cup \ldots \cup F_K = \{1,...,n\}$. Now for each $k = 1,...,K$, we fit our prediction function on all points but those in the kth fold, denoted $\hat{f}^{-(k)}$, and evaluate squared errors on the points in the kth fold,

$$CV_k(\hat{f}^{-(k)}) = \frac{1}{n_k} \sum_{i \in F_k} \left(y_i - \hat{f}^{-(k)}(x_i)\right)^2$$

- Here n_k denotes the number of points in the kth fold, $n_k = |F_k|$. We average these fold-based errors to yield an estimate of the expected test error

$$CV(\hat{f}) = \frac{1}{K} \sum_{k=1}^{K} CV_k(\hat{f}^{-(k)}) \qquad (3)$$

- This is called *K-fold cross-validation*; the special case when $K = n$ is referred to as *leave-one-out cross-validation*.

Cross-validation

- What is the difference between choosing say $K = 5$ (a common choice) versus $K = n$?
  - When K = 5, the function $\hat{f}^{-(k)}$ in each fold k is fit on about $4n/5$ samples, and so we are looking at the errors incurred by a procedure that is trained on less data than the full $\hat{f}$ in (1). Therefore the mean of the CV estimate (3) could be off. When $K = n$, this is not really an issue, since each $\hat{f}^{-(k)}$ is trained on $n - 1$ samples.
  - When $K = n$, the CV estimate (3) is an average of n extremely correlated quantities; this is because each $\hat{f}^{-(k)}$ and $\hat{f}^{-(j)}$ are fit on $n - 2$ common training points. Hence the CV estimate will likely have very high variance. When $K = 5$, the CV estimate will have lower variance, since it the average of quantities that are less correlated, as the fits $\hat{f}^{-(k)}$, $k = 1,...,5$ do not share as much overlapping training data.

- This is tradeoff (the bias-variance tradeoff, in fact!). Usually, a choice like $K = 5$ or $K = 10$ is more common in practice than $K = n$, but this is probably an issue of debate.

Cross-validation

- For K-fold CV, it can be helpful to assign a notion of variability to the CV error estimate.

$$\text{Var}(CV(\hat{f})) = \text{Var}\left(\frac{1}{K}\sum_{k=1}^{K} CV_k(\hat{f}^{-(k)})\right) \approx \frac{1}{K}\text{Var}(CV_1(\hat{f}^{-(1)})) \quad (4)$$

- Why is this an approximation? This would hold exactly if $CV_1(\hat{f}^{-(1)}),\ldots,CV_K(\hat{f}^{-(K)})$ were i.i.d., but they're not.
- This approximation is valid for small K (e.g., K = 5 or 10) but not really for big K (e.g., K = n), because then the quantities $CV_1(\hat{f}^{-(1)}),\ldots,CV_K(\hat{f}^{-(K)})$ are highly correlated.
- For small K (e.g., K = 5 or 10), we can leverage (4) to get an estimate of the variance of the CV error estimate. We just use the sample variance appropriately, so that

$$\frac{1}{K}\sum_{k=1}^{K}\left(CV_k(\hat{f}^{-(k)}) - CV(\hat{f})\right)^2$$

is our estimate of the variance of $CV(\hat{f})$

Cross-validation

- We can use this variance estimate to draw approximate standard deviation bands around a CV error curve, and this leads to model selection heuristics like the *one-standard-error* rule.
- That is, the usual rule for selecting a parameter θ in a family of fitted models $\hat{f}_\theta$, $\theta \in \Theta$ would be minimize the CV error

$$\hat{\theta} = \underset{\theta \in \Theta}{\text{argmin}}\ CV(\hat{f}_\theta) = \frac{1}{K}\sum_{k=1}^{K} CV_k(\hat{f}_\theta^{-(k)})$$

-

Cross-validation

- The one-standard-error rule, instead, chooses the simplest model that is within one standard error (i.e., one standard deviation) of the minimum CV error. How to measure simplicity?
- Well, degrees of freedom is one way to do so! Often, the parametrization $\hat{f}_\theta$ will be such that df($\hat{f}_\theta$) behaves monotonically with θ.
- When df($\hat{f}_\theta$) is monotone increasing with θ, the one-standard-error rule can be written as

$$\tilde{\theta} = \min \left\{ \theta \in \Theta : CV(\hat{f}_\theta) \leq CV(f_{\hat{\theta}}) + SE(f_{\hat{\theta}}) \right\}$$

- Where $SE(\hat{f}_\theta)$ denotes our estimate of the standard deviation of $CV(\hat{f}_\theta)$

$$SE(\hat{f}_\theta) = \frac{1}{\sqrt{K}} \left[\sum_{k=1}^{K} \left(CV_k(\hat{f}_\theta^{-(k)}) - CV(\hat{f}_\theta) \right)^2 \right]^{1/2}$$