Bias/Variance Tradeoff

Rishabh Iyer
University of Texas at Dallas

Acknowledgement: Nick Rouzzi, Vibhav Gogate, David Sontag, Killian Weinberger, Aarti Singh

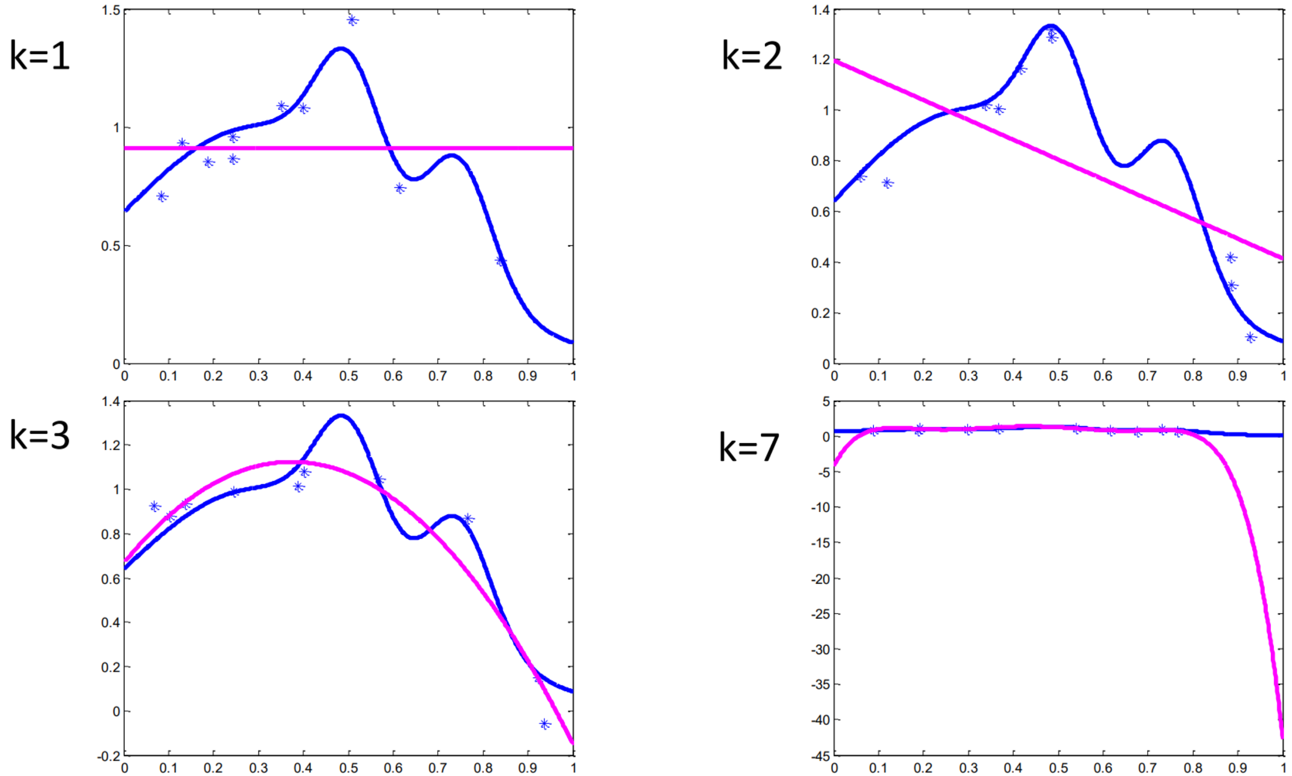
Last Time

- PAC learning
- Bias/variance tradeoff
  - small hypothesis spaces (not enough flexibility) can have high bias
  - rich hypothesis spaces (too much flexibility) can have high variance
- Today: more on this phenomenon and how to get around it

High Variance or Overfitting

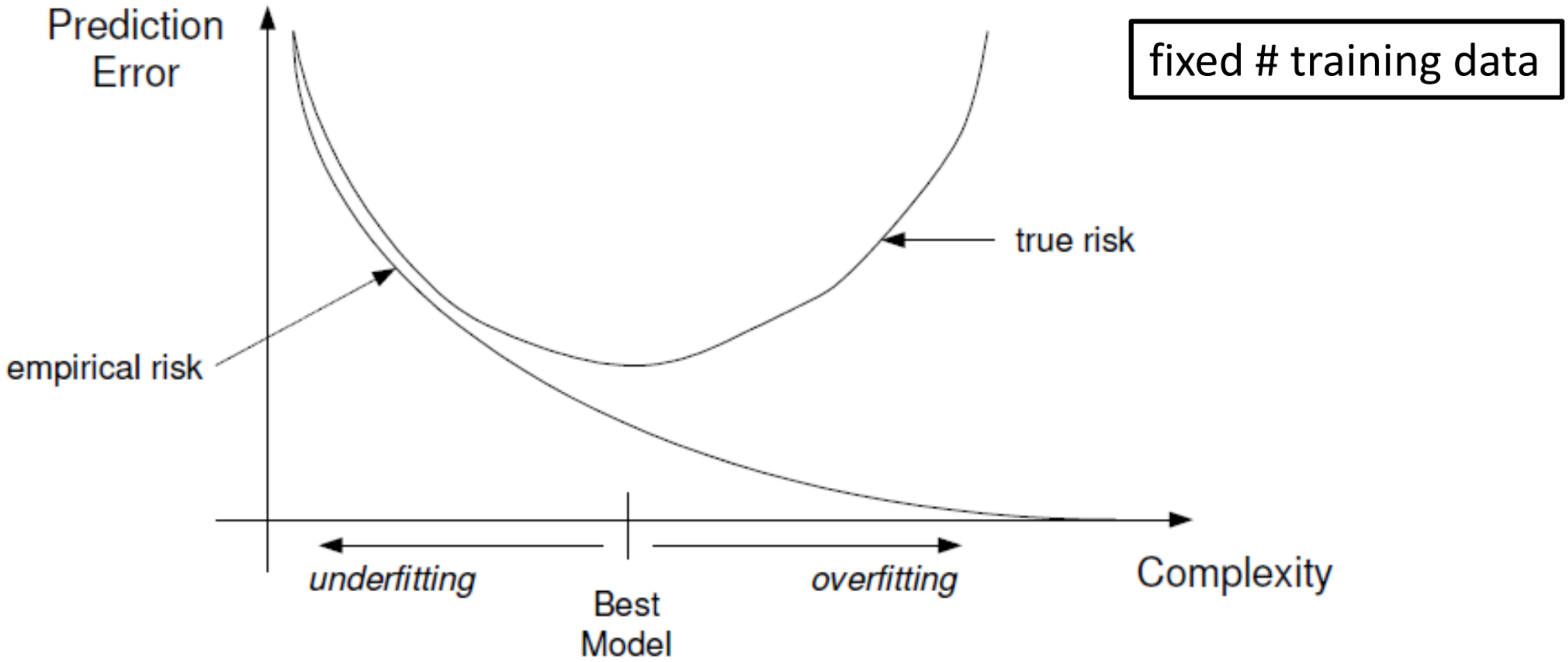
If we allow very complicated predictors, we could overfit the training data.

Examples: Regression (Polynomial of order k – degree up to k-1)



Effect of Model Complexity

If we allow very complicated predictors, we could overfit the training data.



Intuition

- Bias
  - Measures the accuracy or quality of the algorithm
  - High bias means a poor match
- Variance
  - Measures the precision or specificity of the match
  - High variance means a weak match
- We would like to minimize each of these
- Unfortunately, we can’t do this independently, there is a trade-off

Bias-Variance Analysis in Regression

- Dataset: $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$

- True function is $y = f(x) + \epsilon$
  - Where noise, ϵ, is normally distributed with zero mean and standard deviation σ

- Given a set of training examples, $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$, we fit a hypothesis $g(x) = w^T x + b$ to the data to minimize the squared error

$$\sum_i \left[y^{(i)} - g\left(x^{(i)}\right)\right]^2$$

Some Terminology

Expected Label (given $\mathbf{x} \in \mathbb{R}^d$):

$$\bar{y}(\mathbf{x}) = E_{y|\mathbf{x}}\left[Y\right] = \int\limits_y y \Pr(y|\mathbf{x})\partial y.$$

Expected Test Error (given h_D):

$$E_{(\mathbf{x},y)\sim P}\left[\left(h_D(\mathbf{x}) - y\right)^2\right] = \int\limits_x\int\limits_y \left(h_D(\mathbf{x}) - y\right)^2 \Pr(\mathbf{x}, y)\partial y \partial \mathbf{x}.$$

Expected Classifier (given $\mathcal{A}$):

$$\bar{h} = E_{D\sim P^n}\left[h_D\right] = \int h_D \Pr(D)\partial D$$

Expected Test Error (given $\mathcal{A}$):

$$E_{\substack{(\mathbf{x},y)\sim P\\ D\sim P^n}}\left[\left(h_D(\mathbf{x}) - y\right)^2\right] = \int_D \int_{\mathbf{x}} \int_y \left(h_D(\mathbf{x}) - y\right)^2 \mathrm{P}(\mathbf{x}, y)\mathrm{P}(D)\partial \mathbf{x}\partial y \partial D$$

Probability Reminder

- Variance of a random variable, Z

$$Var(Z) = E[(Z - E[Z])^2]$$
$$= E[Z^2 - 2ZE[Z] + E[Z]^2]$$
$$= E[Z^2] - E[Z]^2$$

- Properties of $Var(Z)$

$$Var(aZ) = E[a^2Z^2] - E[aZ]^2 = a^2Var(Z)$$

Bias-Variance-Noise Decomposition

$$
\begin{aligned}
E_{\mathbf{x},y,D}\left[\left[h_D(\mathbf{x}) - y\right]^2\right] &= E_{\mathbf{x},y,D}\left[\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right) + \left(\bar{h}(\mathbf{x}) - y\right)\right]^2\right] \\
&= E_{\mathbf{x},D}\left[\left(\bar{h}_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)^2\right] + 2\ E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x}) - y\right)\right] \\
&\quad + E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - y\right)^2\right]
\end{aligned}
$$

Bias-Variance-Noise Decomposition

$$E_{\mathbf{x},y,D}\left[\left[h_D(\mathbf{x})-y\right]^2\right]=E_{\mathbf{x},y,D}\left[\left[\left(h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right)+\left(\bar{h}(\mathbf{x})-y\right)\right]^2\right]$$

$$=E_{\mathbf{x},D}\left[(\bar{h}_D(\mathbf{x})-\bar{h}(\mathbf{x}))^2\right]+2\ E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x})-y\right)\right]$$

$$+E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-y\right)^2\right]$$

The middle term of the above equation is 0 as we show below

$$\begin{aligned}
E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x})-y\right)\right] &= E_{\mathbf{x},y}\left[E_D\left[h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right]\left(\bar{h}(\mathbf{x})-y\right)\right] \\
&= E_{\mathbf{x},y}\left[\left(E_D\left[h_D(\mathbf{x})\right]-\bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x})-y\right)\right] \\
&= E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x})-y\right)\right] \\
&= E_{\mathbf{x},y}\left[0\right] \\
&= 0
\end{aligned}$$

Adapted from Killian Weinberger (Cornell)

Bias-Variance-Noise Decomposition

Returning to the earlier expression, we're left with the variance and another term

$$E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x}) - y\right)^2\right] = \underbrace{E_{\mathbf{x},D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)^2\right]}_{\text{Variance}} + E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - y\right)^2\right]$$

Adapted from Killian Weinberger (Cornell)

Bias-Variance-Noise Decomposition

Returning to the earlier expression, we're left with the variance and another term

$$E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x})-y\right)^2\right] = \underbrace{E_{\mathbf{x},D}\left[\left(h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right)^2\right]}_{\text{Variance}} + E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-y\right)^2\right]$$

We can break down the second term in the above equation as follows:

$$\begin{aligned} E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-y\right)^2\right] &= E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)+\left(\bar{y}(\mathbf{x})-y\right)^2\right] \\ &= \underbrace{E_{\mathbf{x},y}\left[\left(\bar{y}(\mathbf{x})-y\right)^2\right]}_{\text{Noise}} + \underbrace{E_{\mathbf{x}}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)^2\right]}_{\text{Bias}^2} + 2\ E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\left(\bar{y}(\mathbf{x})-y\right)\right] \end{aligned}$$

Adapted from Killian Weinberger (Cornell)

Bias-Variance-Noise Decomposition

Returning to the earlier expression, we're left with the variance and another term

$$E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x}) - y\right)^2\right] = \underbrace{E_{\mathbf{x},D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)^2\right]}_{\text{Variance}} + E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - y\right)^2\right]$$

We can break down the second term in the above equation as follows:

$$\begin{aligned} E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - y\right)^2\right] &= E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right) + \left(\bar{y}(\mathbf{x}) - y\right)^2\right] \\ &= \underbrace{E_{\mathbf{x},y}\left[\left(\bar{y}(\mathbf{x}) - y\right)^2\right]}_{\text{Noise}} + \underbrace{E_{\mathbf{x}}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)^2\right]}_{\text{Bias}^2} + 2\ E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)\left(\bar{y}(\mathbf{x}) - y\right)\right] \end{aligned}$$

The third term in the equation above is 0

Adapted from Killian Weinberger (Cornell)

Bias-Variance-Noise Decomposition

The third term in the equation above is 0, as we show below

$$\begin{aligned}
E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)\left(\bar{y}(\mathbf{x}) - y\right)\right] &= E_{\mathbf{x}}\left[E_{y|\mathbf{x}}\left[\bar{y}(\mathbf{x}) - y\right]\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)\right] \\
&= E_{\mathbf{x}}\left[E_{y|\mathbf{x}}\left[\bar{y}(\mathbf{x}) - y\right]\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)\right] \\
&= E_{\mathbf{x}}\left[\left(\bar{y}(\mathbf{x}) - E_{y|\mathbf{x}}\left[y\right]\right)\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)\right] \\
&= E_{\mathbf{x}}\left[\left(\bar{y}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)\right] \\
&= E_{\mathbf{x}}\left[0\right] \\
&= 0
\end{aligned}$$

This gives us the decomposition of expected test error as follows

$$\underbrace{E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x}) - y\right)^2\right]}_{\text{Expected Test Error}} = \underbrace{E_{\mathbf{x},D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)^2\right]}_{\text{Variance}} + \underbrace{E_{\mathbf{x},y}\left[\left(\bar{y}(\mathbf{x}) - y\right)^2\right]}_{\text{Noise}} + \underbrace{E_{\mathbf{x}}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)^2\right]}_{\text{Bias}^2}$$

Adapted from Killian Weinberger (Cornell)

Bias, Variance, and Noise

This gives us the decomposition of expected test error as follows

$$\underbrace{E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x}) - y\right)^2\right]}_{\mathrm{Expected\ Test\ Error}} = \underbrace{E_{\mathbf{x},D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)^2\right]}_{\mathrm{Variance}} + \underbrace{E_{\mathbf{x},y}\left[\left(\bar{y}(\mathbf{x}) - y\right)^2\right]}_{\mathrm{Noise}}$$

$$+ \underbrace{E_{\mathbf{x}}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)^2\right]}_{\mathrm{Bias}^2}$$

Variance: Captures how much your classifier changes if you train on a different training set. How "over-specialized" is your classifier to a particular training set (overfitting)? If we have the best possible model for our training data, how far off are we from the average classifier?

Bias: What is the inherent error that you obtain from your classifier even with infinite training data? This is due to your classifier being "biased" to a particular kind of solution (e.g. linear classifier). In other words, bias is inherent to your model.

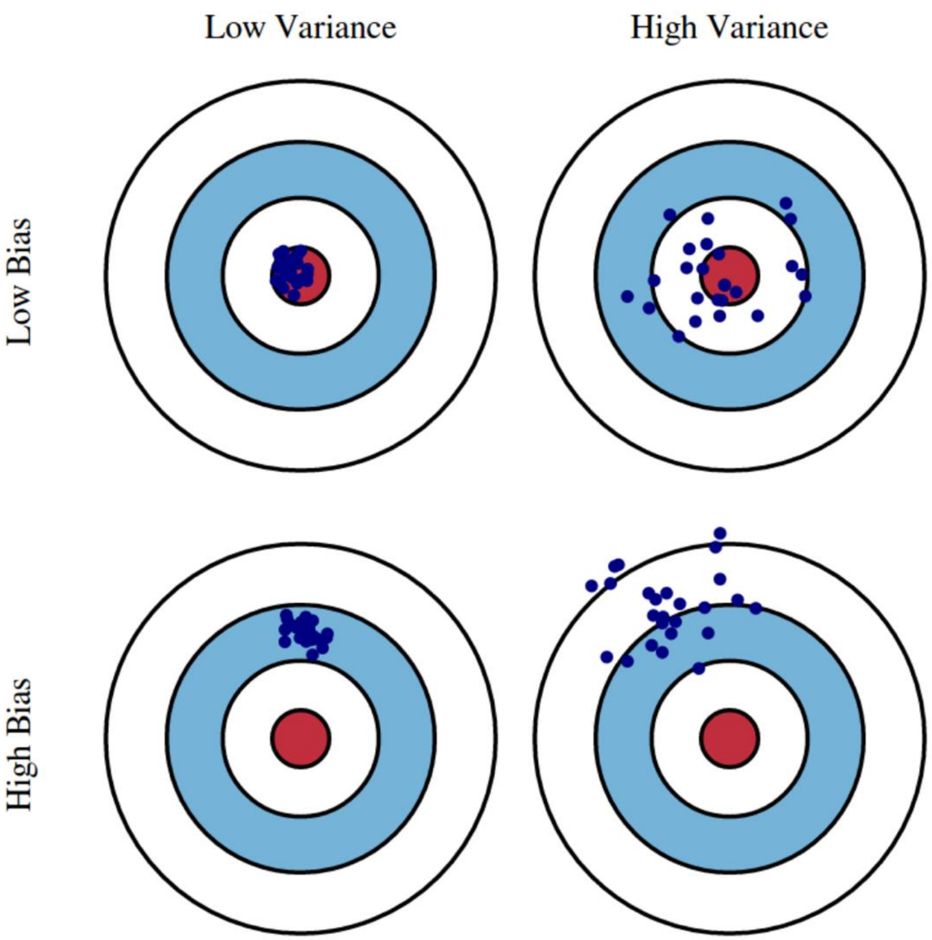
Noise: How big is the data-intrinsic noise? This error measures ambiguity due to your data distribution and feature representation. You can never beat this, it is an aspect of the data.

Bias, Variance, and Noise

$$\underbrace{E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x}) - y\right)^2\right]}_{\text{Expected Test Error}} = \underbrace{E_{\mathbf{x},D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)^2\right]}_{\text{Variance}}$$

$$+ \underbrace{E_{\mathbf{x},y}\left[\left(\bar{y}(\mathbf{x}) - y\right)^2\right]}_{\text{Noise}}$$

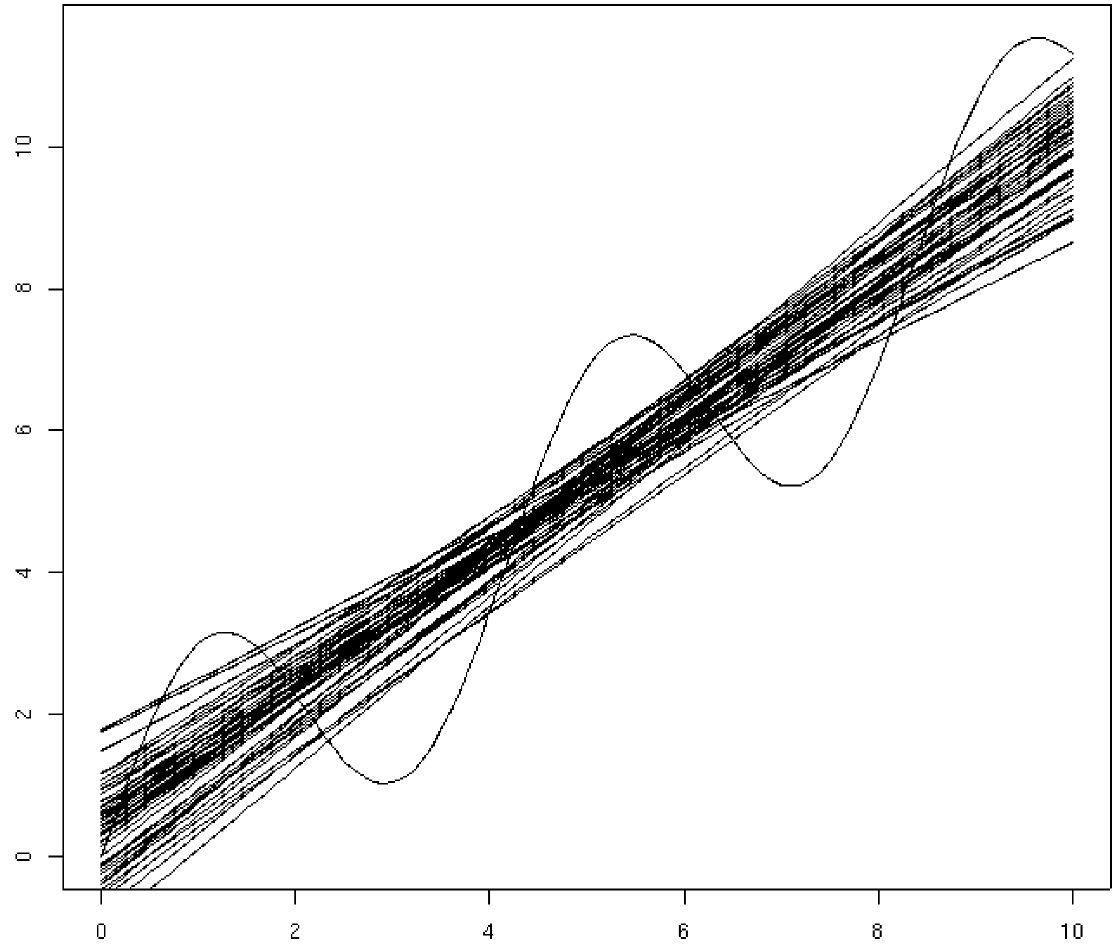
$$+ \underbrace{E_{\mathbf{x}}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)^2\right]}_{\text{Bias}^2}$$



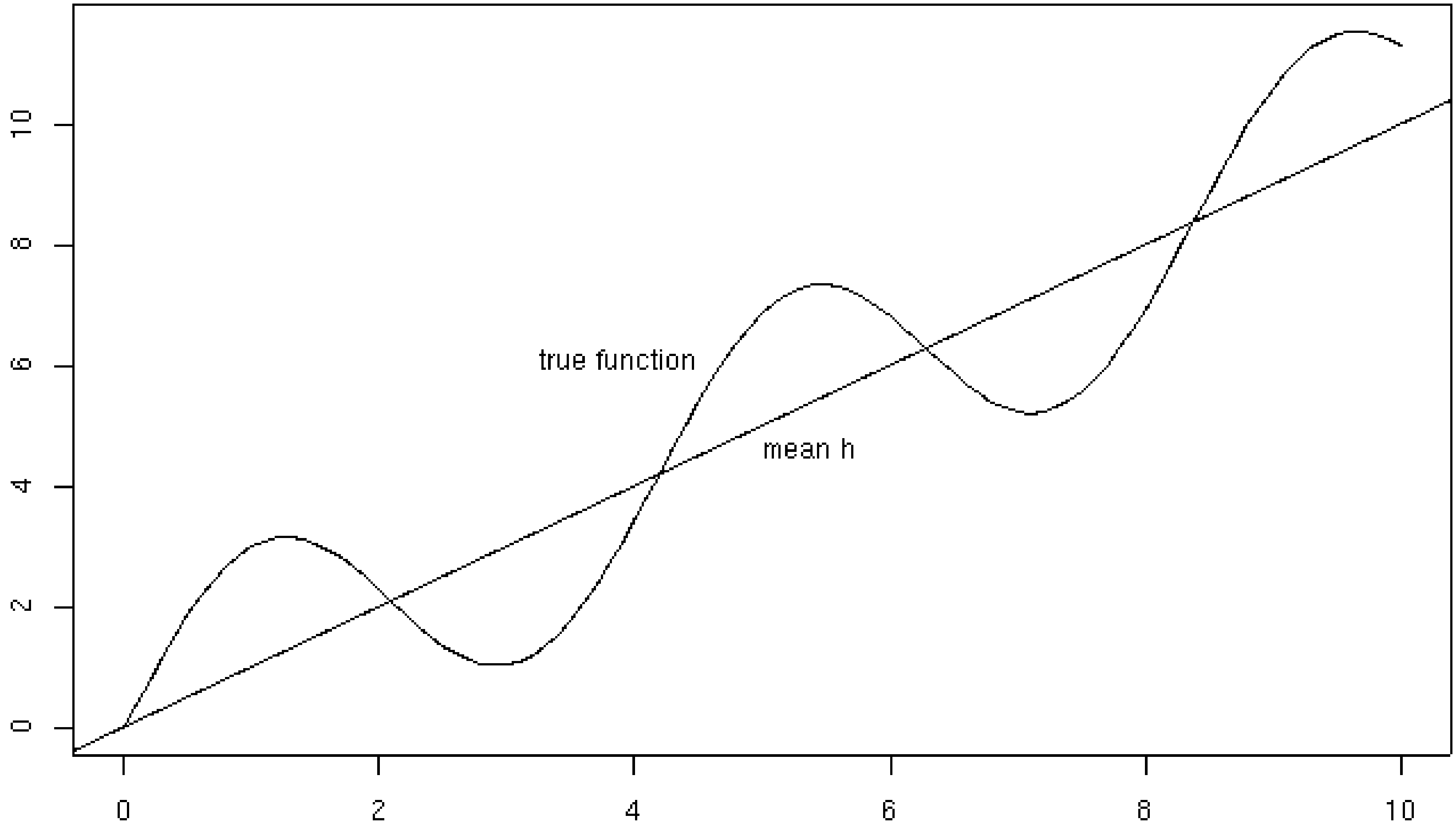
Adapted from Killian Weinberger (Cornell)

2-D Example

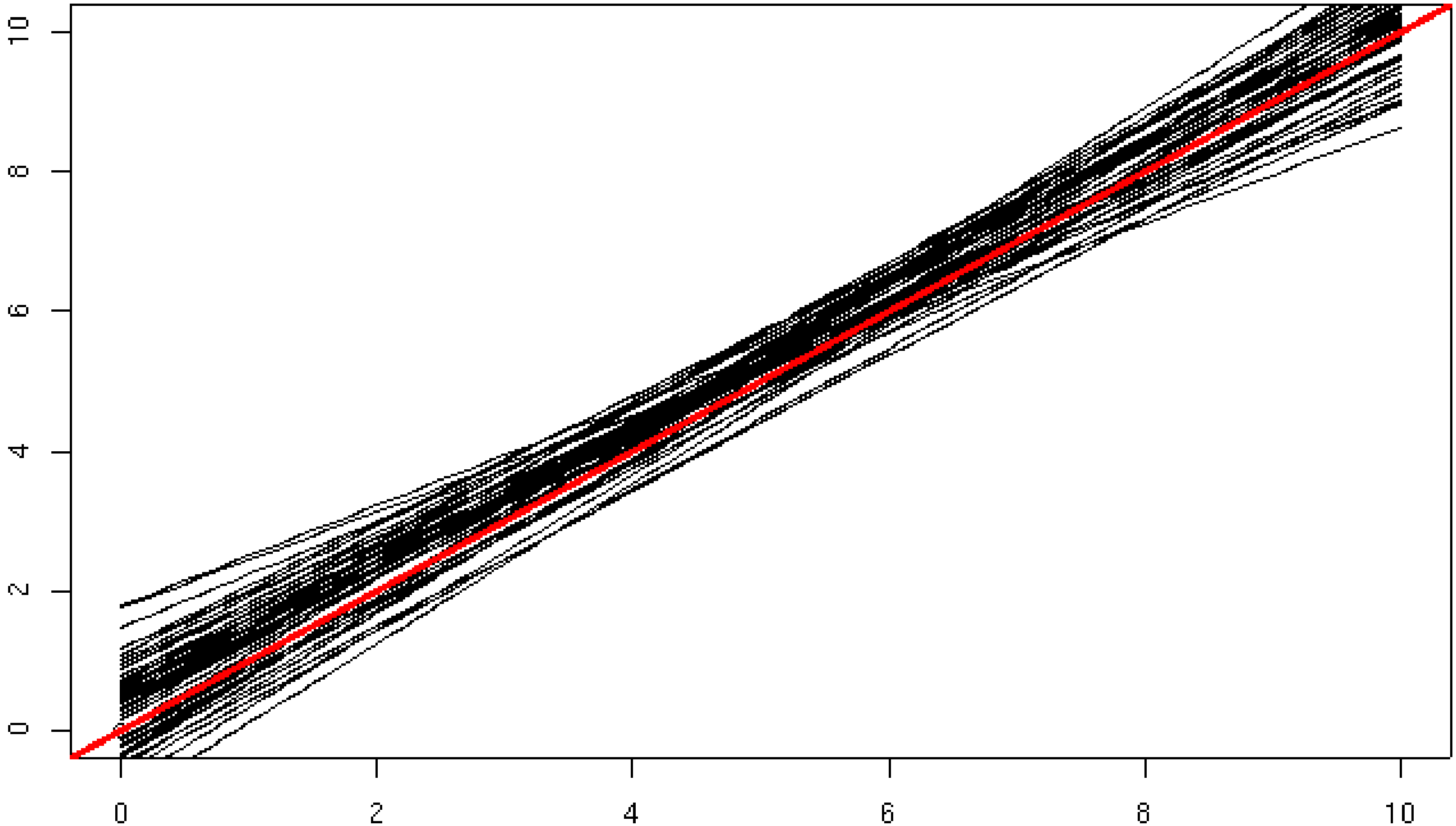
50 fits (20 examples each)



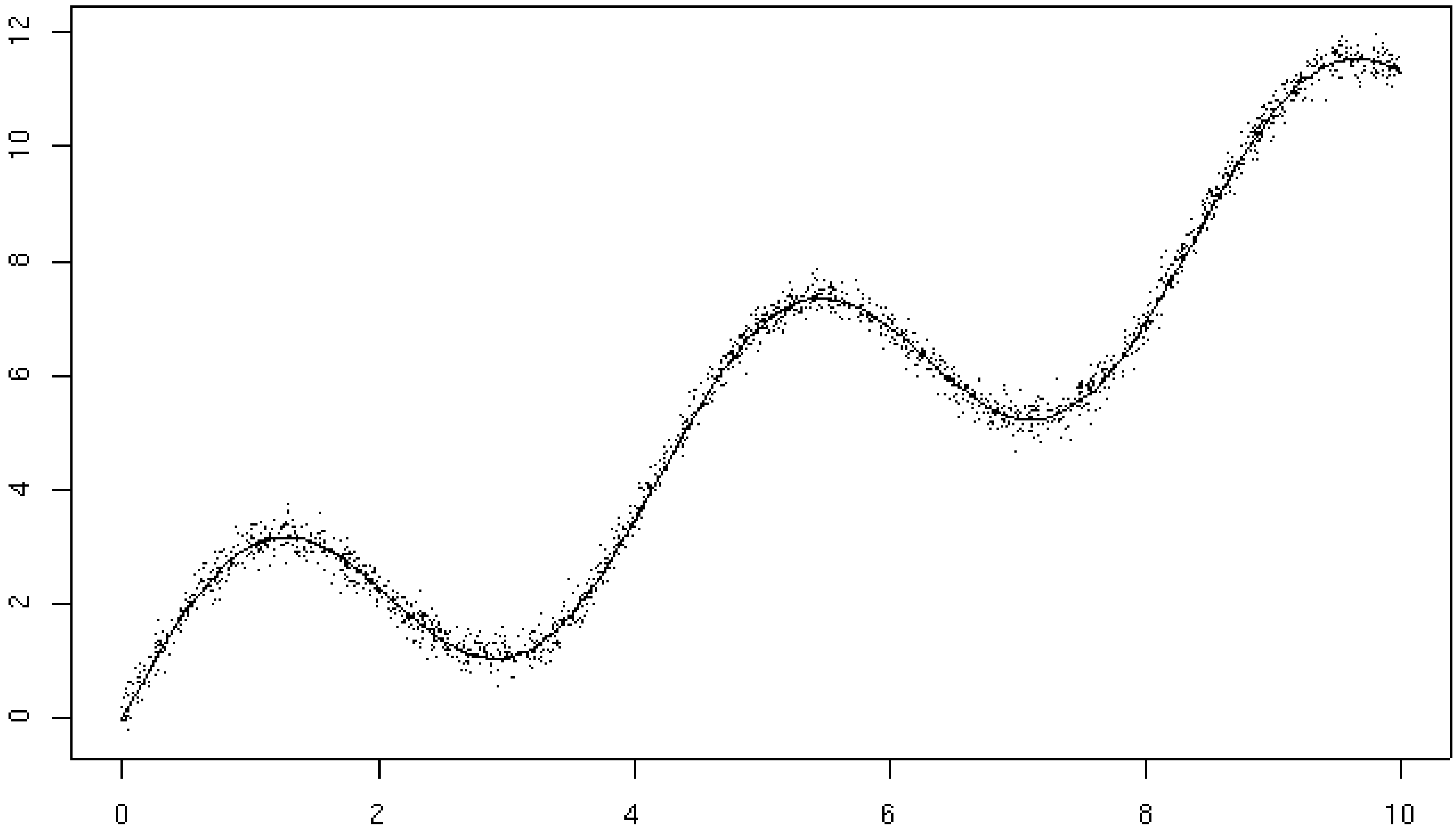
Bias



Variance

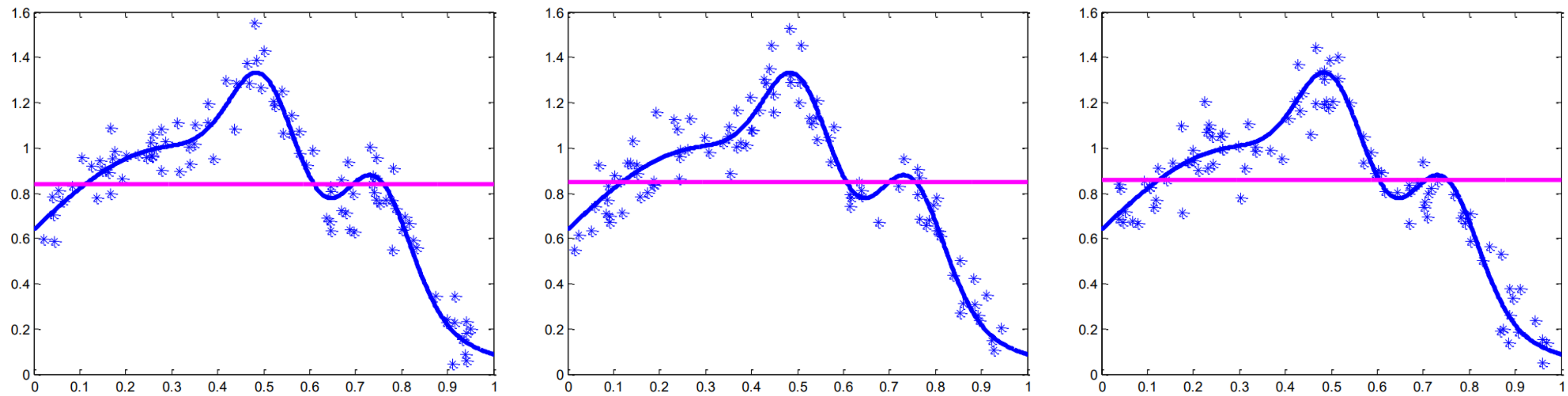


Noise

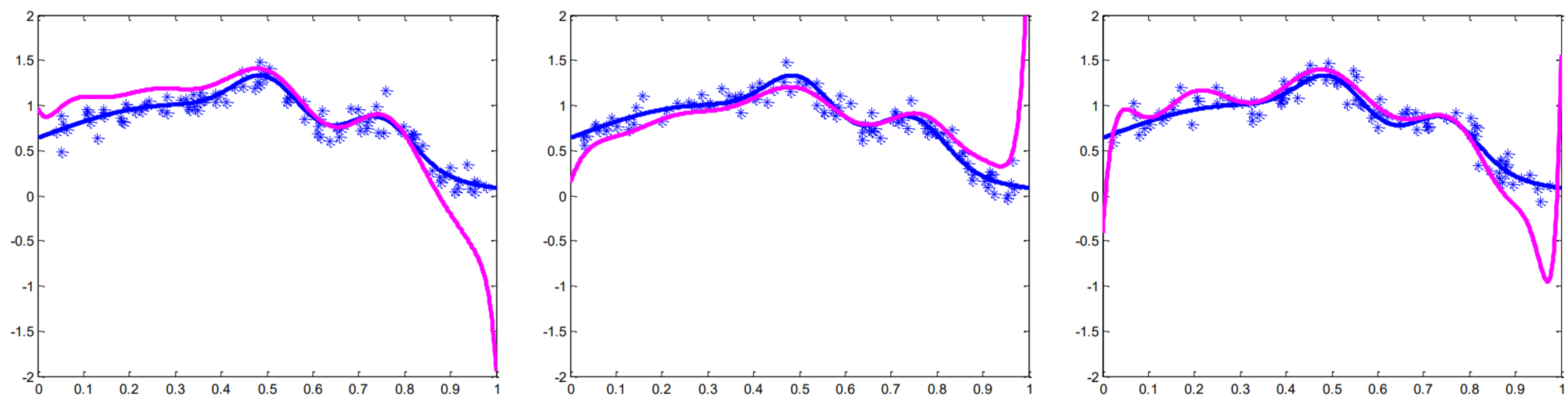


Bias-Variance Tradeoff

Large bias, Small variance – poor approximation but robust/stable



Small bias, Large variance – good approximation but instable



Bias

- Low bias
  - ?
- High bias
  - ?

Bias

- Low bias
  - Linear regression applied to linear data
  - 2nd degree polynomial applied to quadratic data
- High bias
  - Constant function applied to non-constant data
  - Linear regression applied to highly non-linear data

Variance

- Low variance
  - ?
- High variance
  - ?

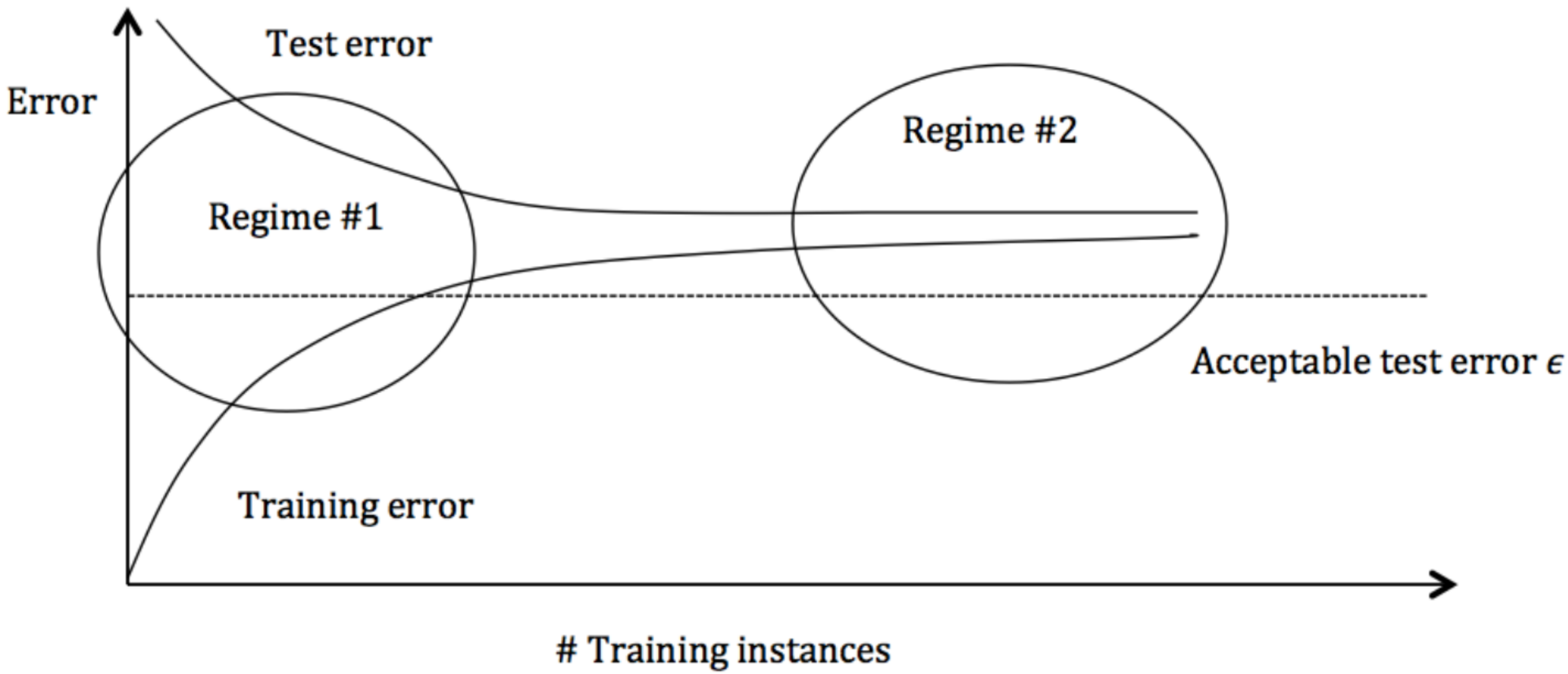
Variance

- Low variance
  - Constant function
  - Model independent of training data
- High variance
  - High degree polynomial

Bias/Variance Tradeoff

- (bias^2+variance) is what counts for prediction
- As we saw in PAC learning, we often have
  - Low bias $\Rightarrow$ high variance
  - Low variance $\Rightarrow$ high bias
  - How can we deal with this in practice?

Detecting High Variance/Bias



Adapted from Killian Weinberger (Cornell)

Detecting High Variance

Regime 1 (High Variance)

In the first regime, the cause of the poor performance is high variance.

Symptoms:

1. Training error is much lower than test error
2. Training error is lower than ϵ
3. Test error is above ϵ

Remedies:

- Add more training data
- Reduce model complexity -- complex models are prone to high variance
- Bagging (will be covered later in the course)

Adapted from Killian Weinberger (Cornell)

Detecting High Bias

Regime 2 (High Bias)

Unlike the first regime, the second regime indicates high bias: the model being used is not robust enough to produce an accurate prediction.

Symptoms:

1. Training error is higher than ϵ

Remedies:

- Use more complex model (e.g. kernelize, use non-linear models)
- Add features
- Boosting (will be covered later in the course)

Adapted from Killian Weinberger (Cornell)

How to select the right model?

Model Spaces with increasing complexity:

- Nearest-Neighbor classifiers with varying neighborhood sizes k = 1,2,3,…
  Small neighborhood => Higher complexity

- Decision Trees with depth k or with k leaves
  Higher depth/ More # leaves => Higher complexity

- Regression with polynomials of order k = 0, 1, 2, …
  Higher degree => Higher complexity

- Kernel Regression with bandwidth h
  Small bandwidth => Higher complexity

How can we select the right complexity model ?

Held out Validation Set

We would like to pick the model that has smallest generalization error.

Can judge generalization error by using an independent sample of data.

Hold - out procedure:

n data points available $D \equiv \{X_i, Y_i\}_{i=1}^n$

1) Split into two sets: Training dataset $D_T = \{X_i, Y_i\}_{i=1}^m$ Validation dataset $D_V = \{X_i, Y_i\}_{i=m+1}^n$ NOT test Data !!

2) Use D_T for training a predictor from each model class:

$$\hat{f}_\lambda = \arg\min_{f \in \mathcal{F}_\lambda} \hat{R}_T(f)$$

Evaluated on training dataset D_T

Held out Validation Set

3) Use *Dv* to select the model class which has smallest empirical error on D_V

$$\widehat{\lambda} = \arg\min_{\lambda \in \Lambda} \widehat{R}_V(\widehat{f}_\lambda)$$

Evaluated on validation dataset D_V

4) Hold-out predictor

$$\widehat{f} = \widehat{f}_{\widehat{\lambda}}$$

Intuition: Small error on one set of data will not imply small error on a randomly sub-sampled second set of data

Ensures method is "stable"

Cross Validation

K-fold cross-validation

Create K-fold partition of the dataset.
Form K hold-out predictors, each time using one partition as validation and rest K-1 as training datasets.
Final predictor is average/majority vote over the K hold-out estimates.