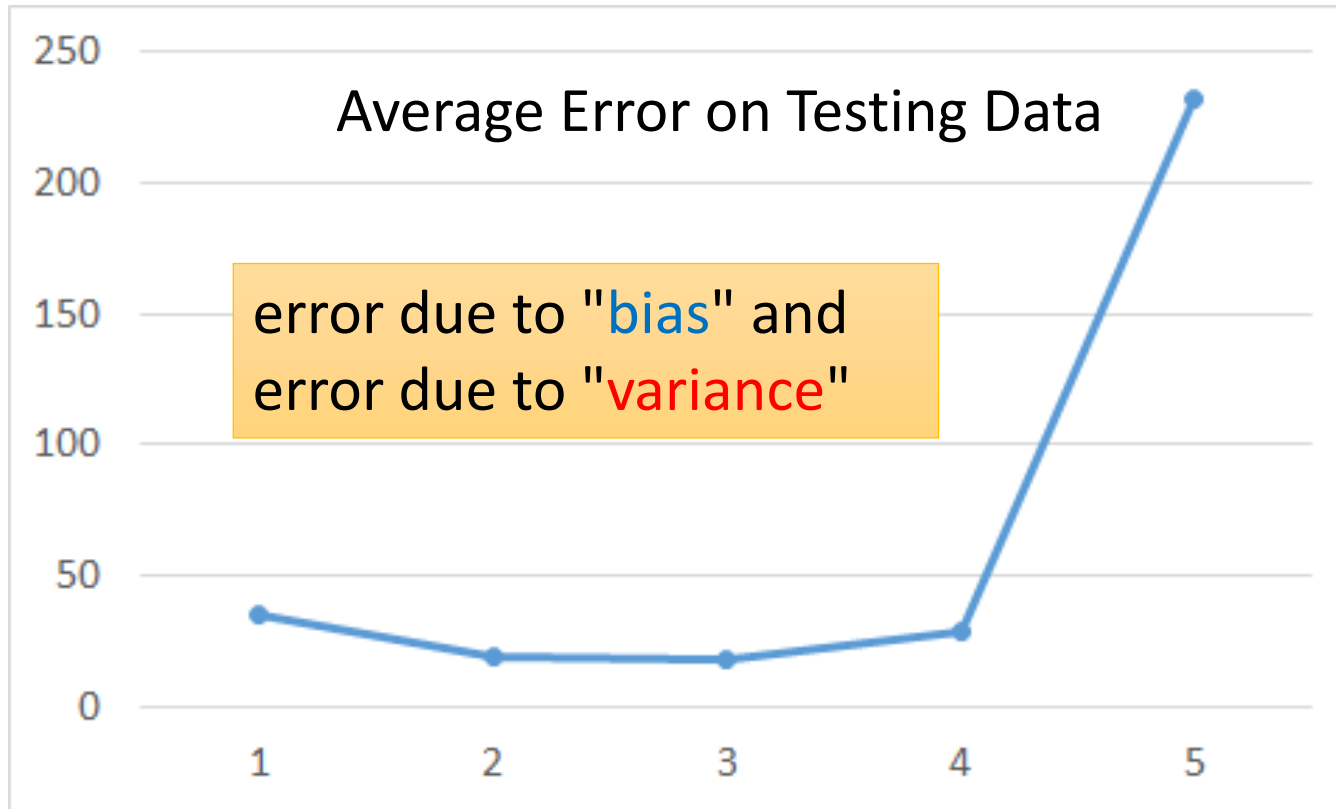


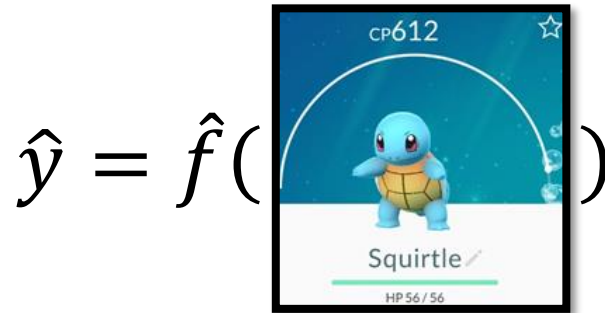
Where does the error
come from?

Review



A more complex model does not always lead to better performance on testing data.

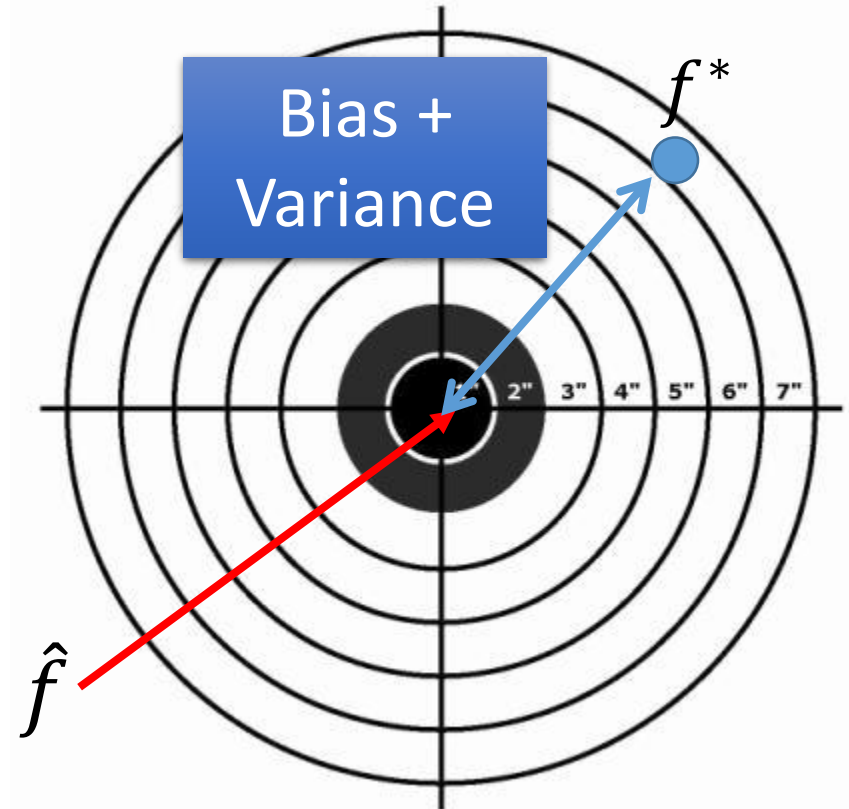
Estimator



Only Niantic knows \hat{f}

From training data,
we find f^*

f^* is an estimator of \hat{f}



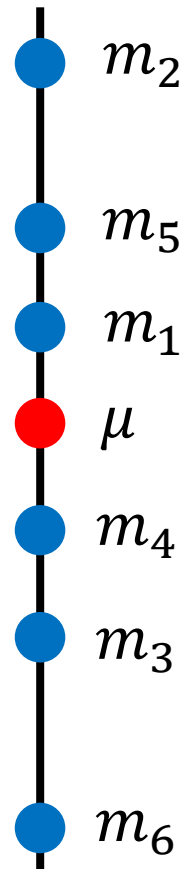
Bias and Variance of Estimator

- Estimate the mean of a variable x
 - assume the mean of x is μ
 - assume the variance of x is σ^2
- Estimator of mean μ
 - Sample N points: $\{x^1, x^2, \dots, x^N\}$

$$m = \frac{1}{N} \sum_n x^n \neq \mu$$

$$E[m] = E \left[\frac{1}{N} \sum_n x^n \right] = \frac{1}{N} \sum_n E[x^n] = \mu$$

unbiased



Bias and Variance of Estimator

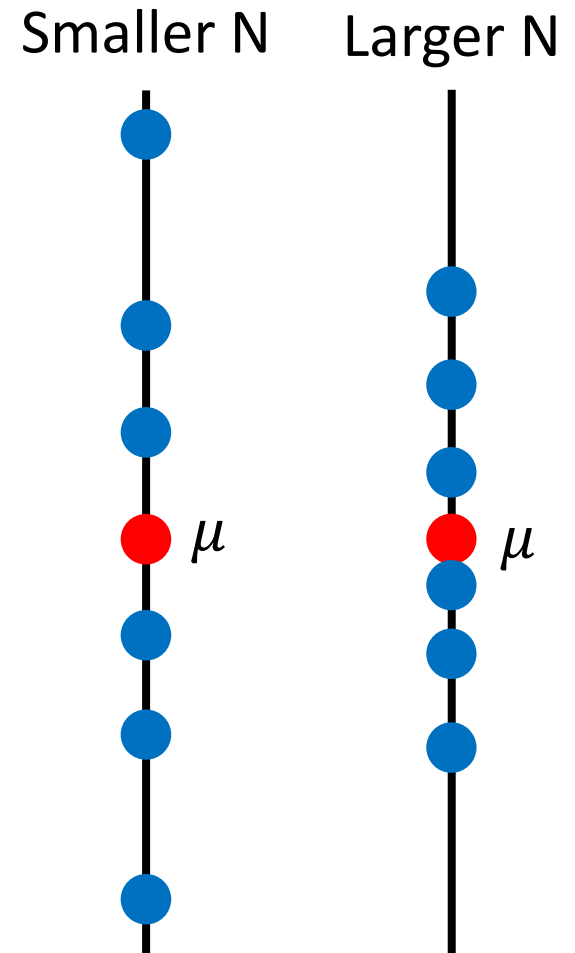
- Estimate the mean of a variable x
 - assume the mean of x is μ
 - assume the variance of x is σ^2
- Estimator of mean μ
 - Sample N points: $\{x^1, x^2, \dots, x^N\}$

$$m = \frac{1}{N} \sum_n x^n \neq \mu$$

$$\text{Var}[m] = \frac{\sigma^2}{N}$$

Variance depends on the number of samples

unbiased



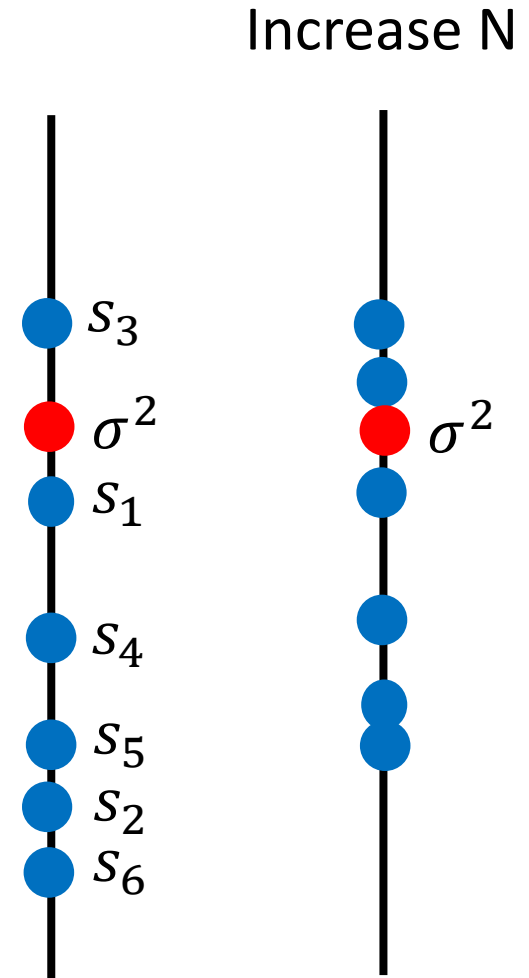
Bias and Variance of Estimator

- Estimate the mean of a variable x
 - assume the mean of x is μ
 - assume the variance of x is σ^2
- Estimator of variance σ^2
 - Sample N points: $\{x^1, x^2, \dots, x^N\}$

$$m = \frac{1}{N} \sum_n x^n \quad s = \frac{1}{N} \sum_n (x^n - m)^2$$

Biased estimator

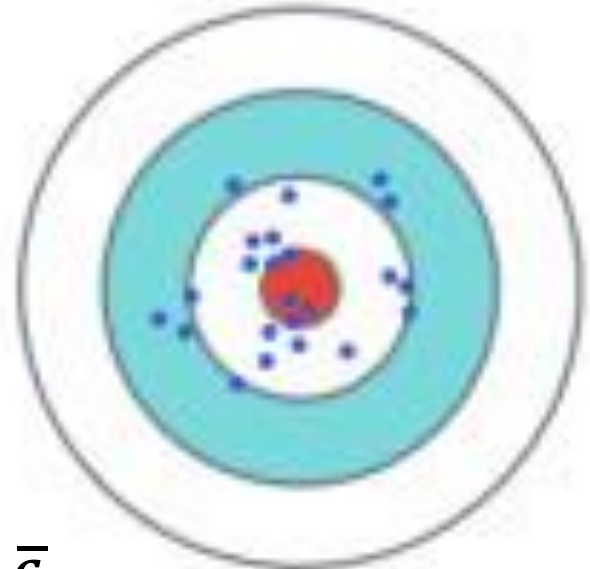
$$E[s] = \frac{N-1}{N} \sigma^2 \neq \sigma^2$$



Low Variance

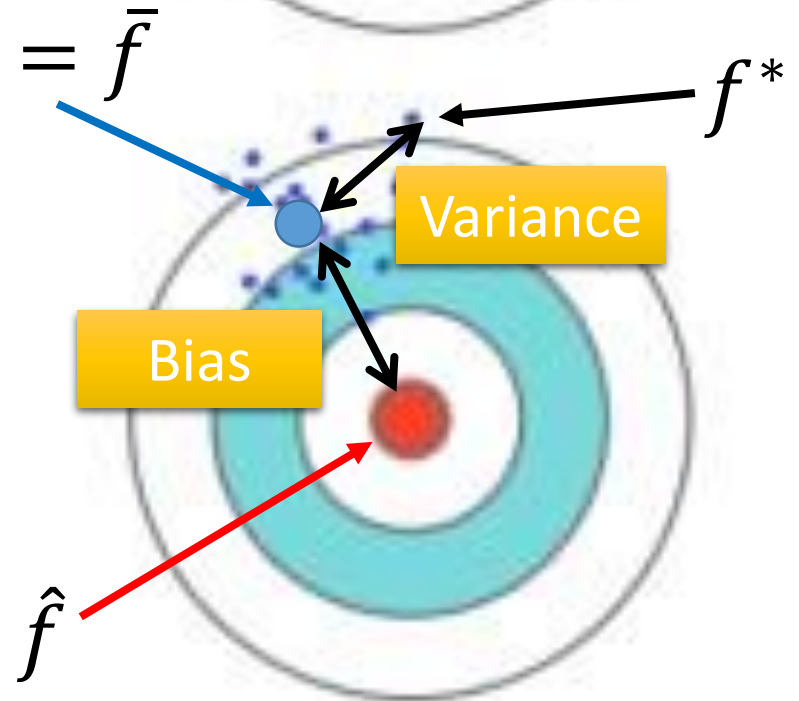
High Variance

Low Bias



$$E[f^*] = \bar{f}$$

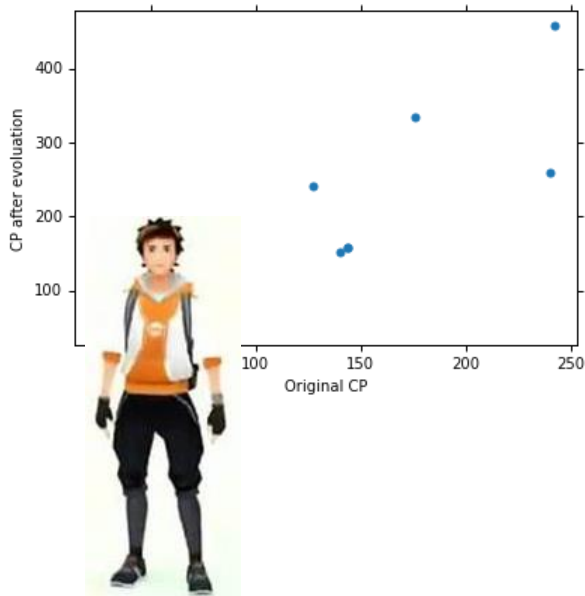
High Bias



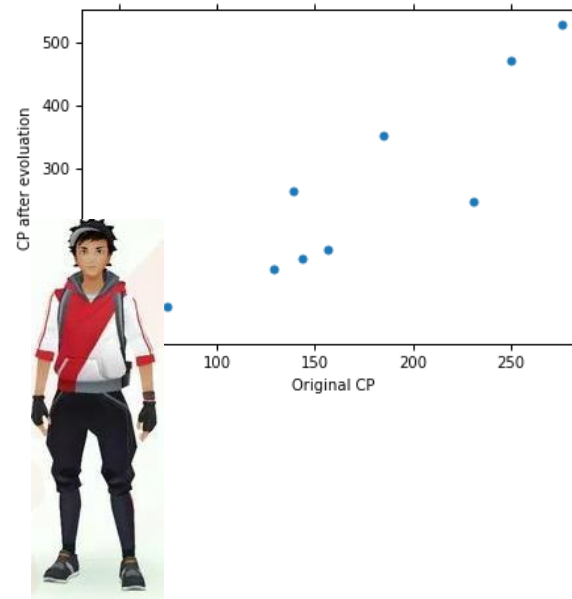
Parallel Universes

- In all the universes, we are collecting (catching) 10 Pokémon as training data to find f^*

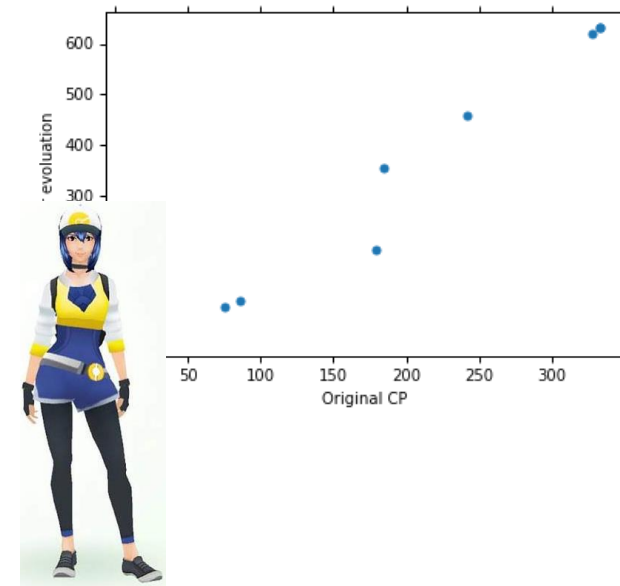
Universe 1



Universe 2



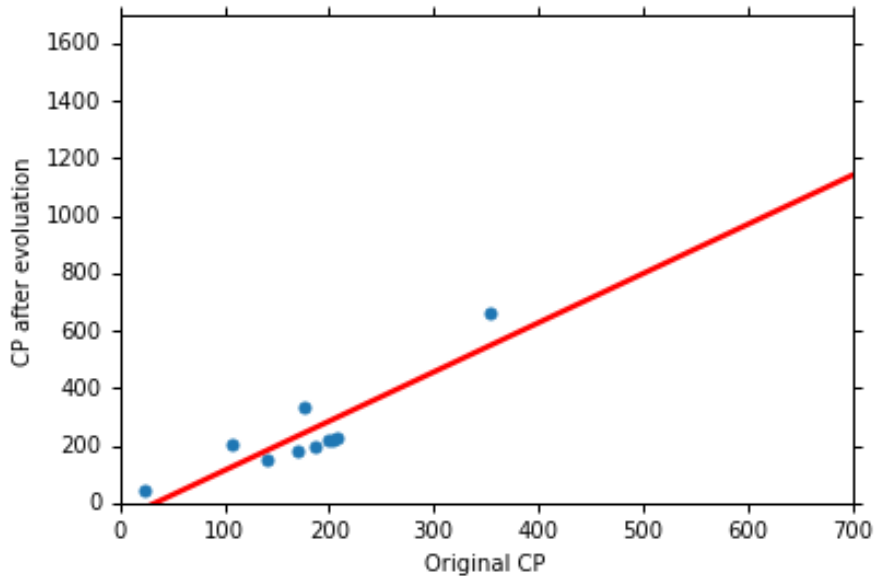
Universe 3



Parallel Universes

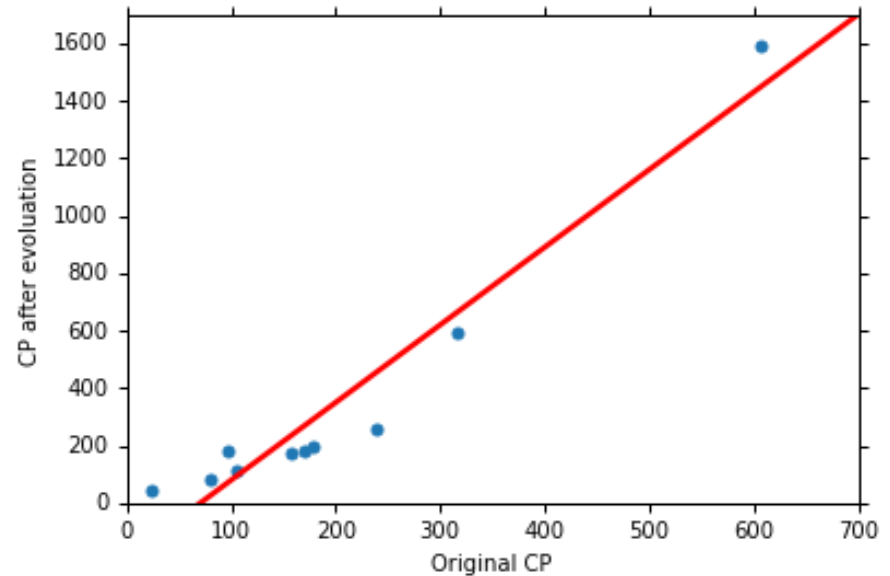
- In different universes, we use the same model, but obtain different f^*

Universe 123



$$y = b + w \cdot x_{cp}$$

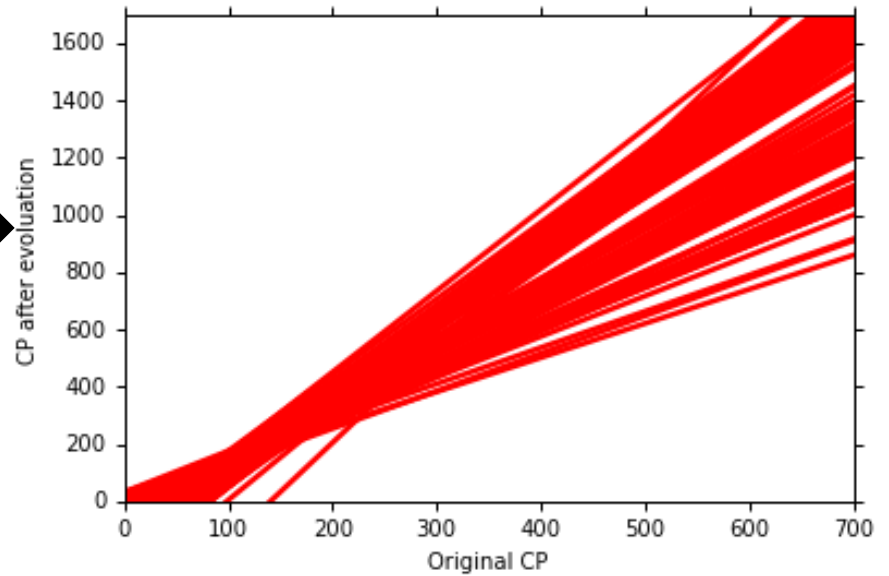
Universe 345



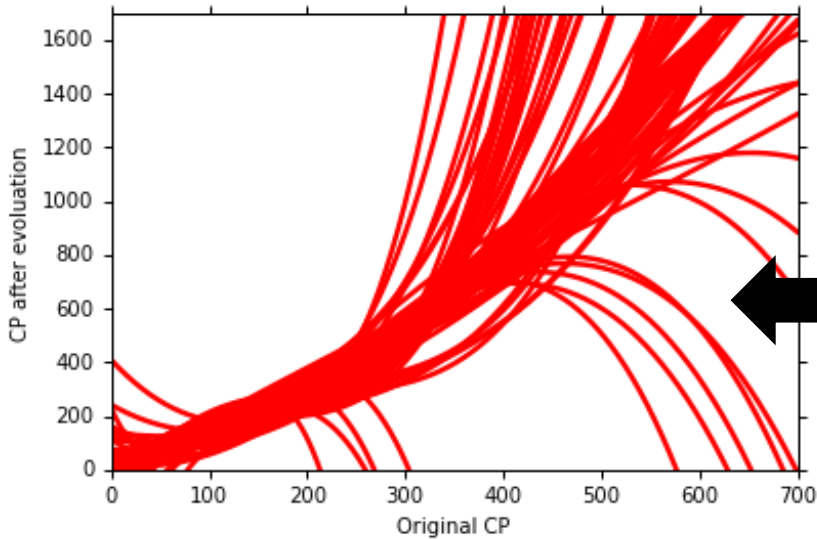
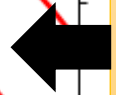
$$y = b + w \cdot x_{cp}$$

f^* in 100 Universes

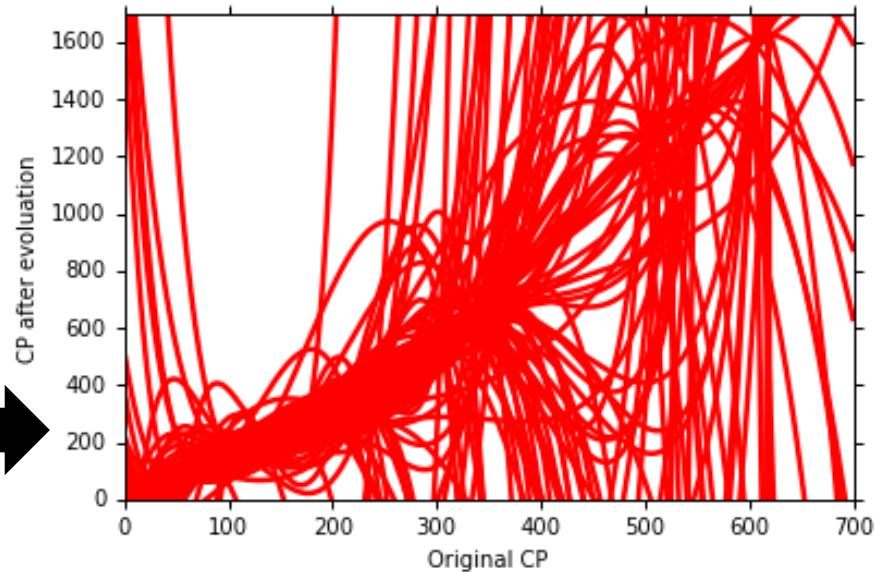
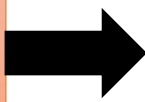
$$y = b + w \cdot x_{cp}$$



$$y = b + w_1 \cdot x_{cp} + w_2 \cdot (x_{cp})^2 + w_3 \cdot (x_{cp})^3$$

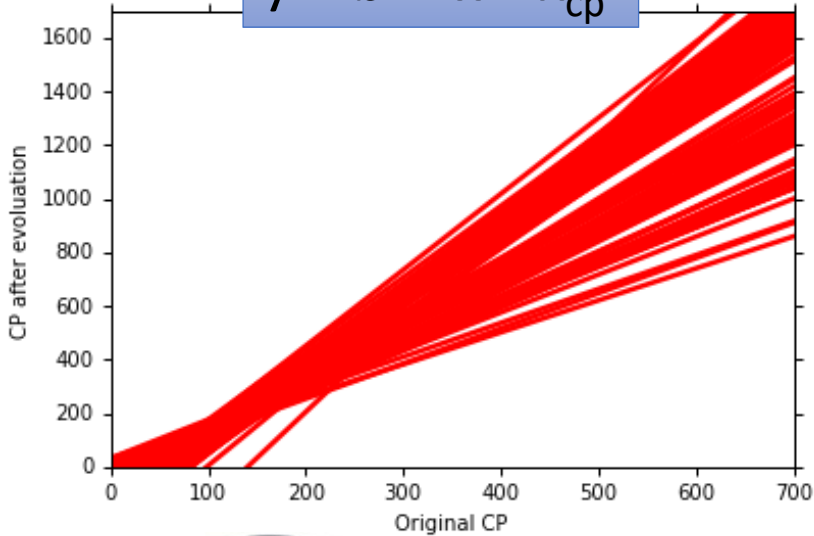


$$y = b + w_1 \cdot x_{cp} + w_2 \cdot (x_{cp})^2 + w_3 \cdot (x_{cp})^3 + w_4 \cdot (x_{cp})^4 + w_5 \cdot (x_{cp})^5$$



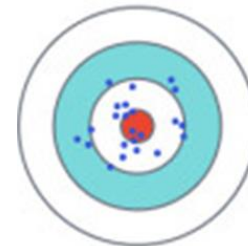
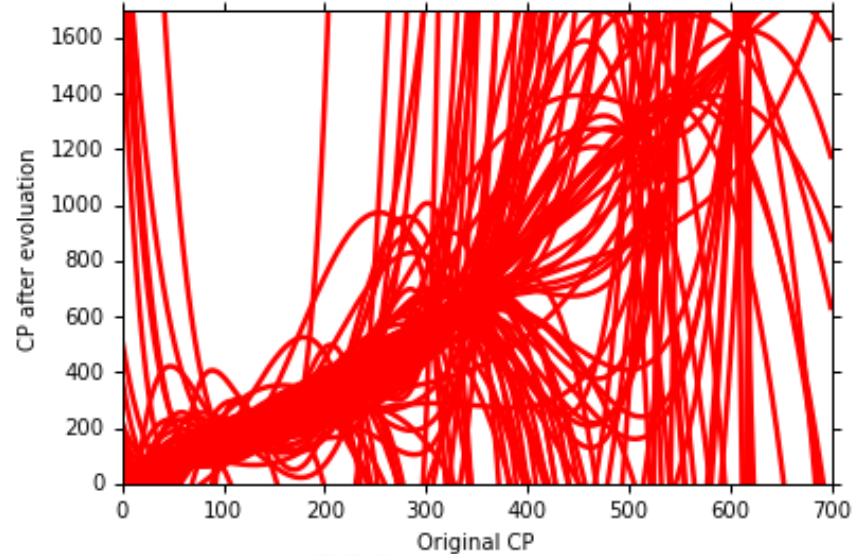
Variance

$$y = b + w \cdot x_{cp}$$



Small
Variance

$$y = b + w_1 \cdot x_{cp} + w_2 \cdot (x_{cp})^2 + w_3 \cdot (x_{cp})^3 + w_4 \cdot (x_{cp})^4 + w_5 \cdot (x_{cp})^5$$



Large
Variance

Simpler model is less influenced by the sampled data

Consider the extreme case $f(x) = 5$

Bias

$$E[f^*] = \bar{f}$$

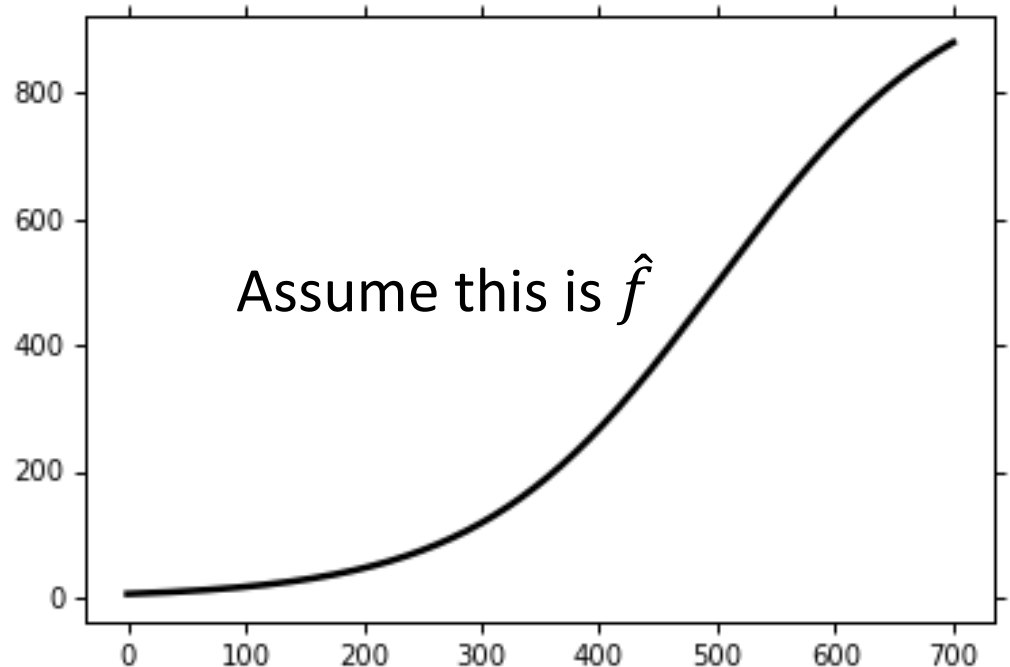
- Bias: If we average all the f^* , is it close to \hat{f} ?



Large
Bias



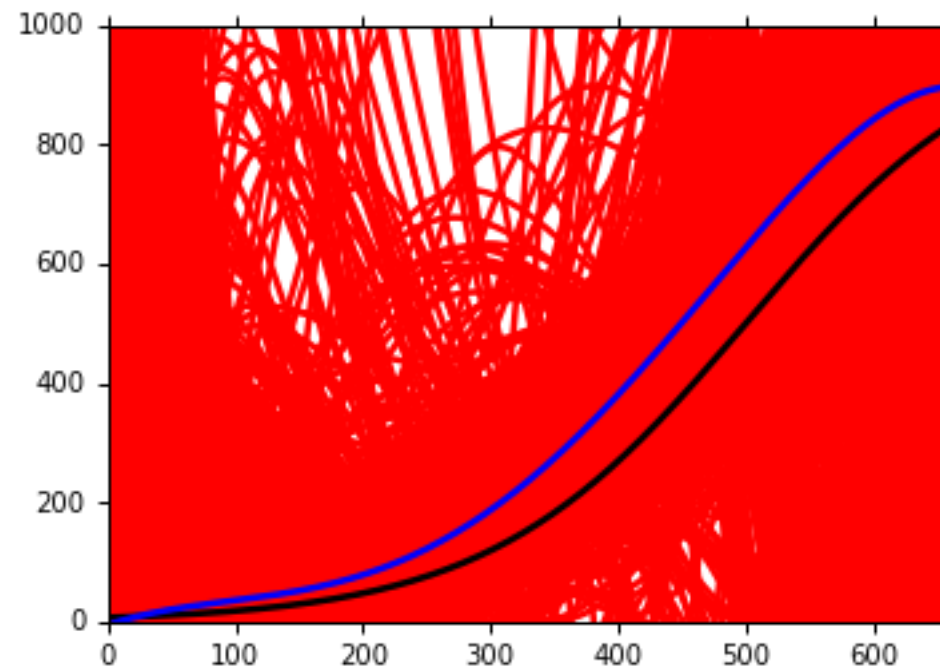
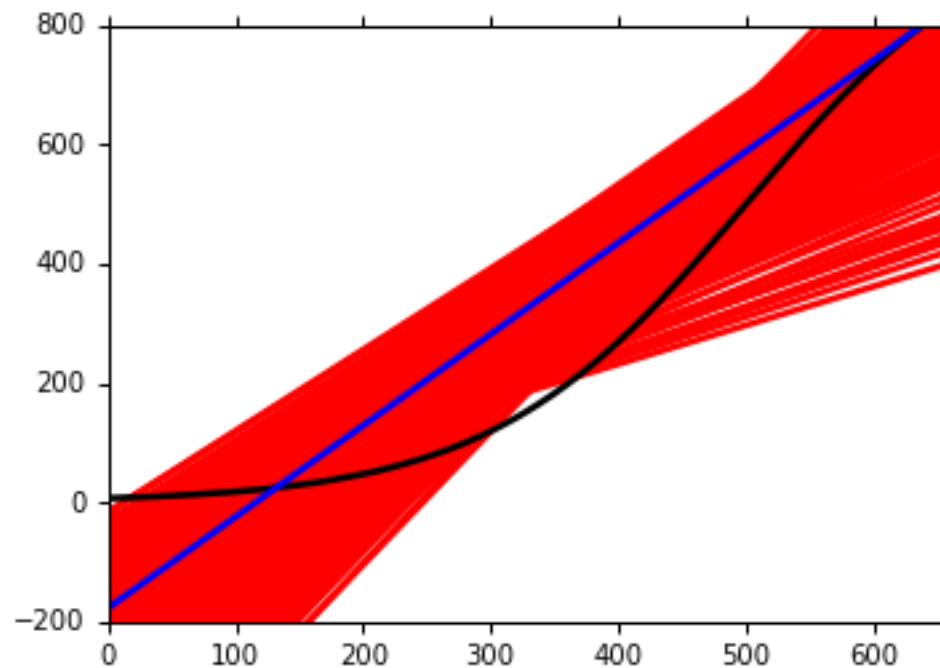
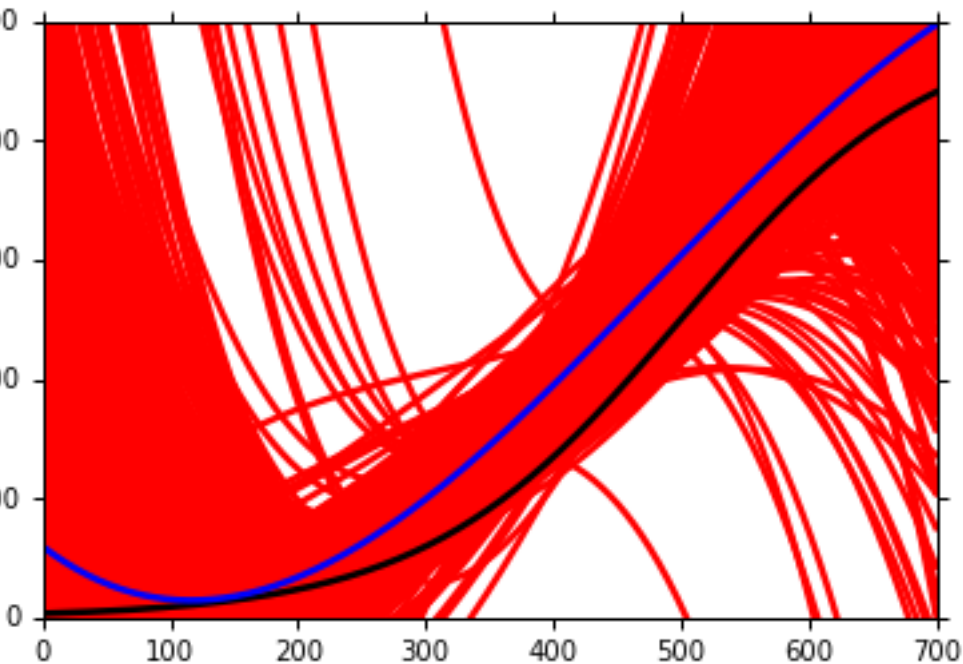
Small
Bias



Black curve: the true function \hat{f}

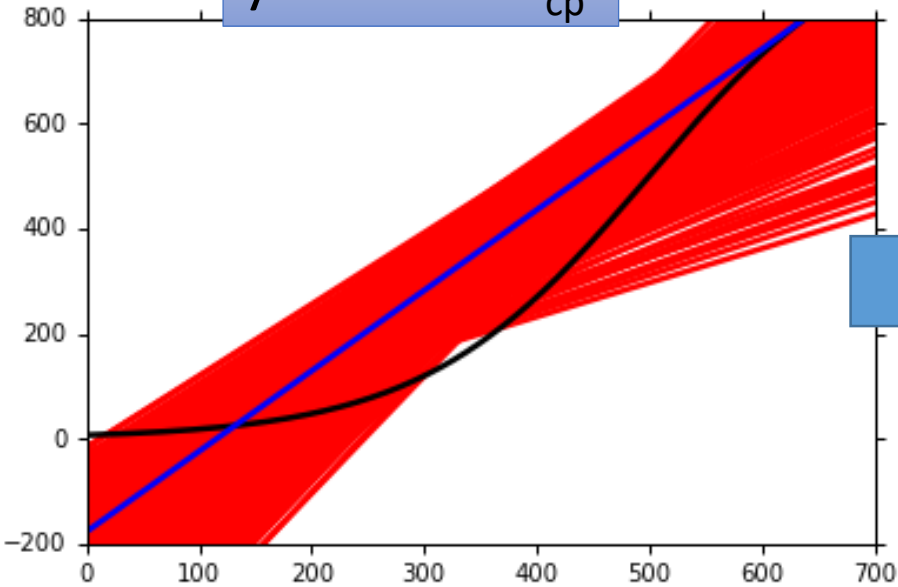
Red curves: 5000 f^*

Blue curve: the average of 5000 f^*
 $= \bar{f}$

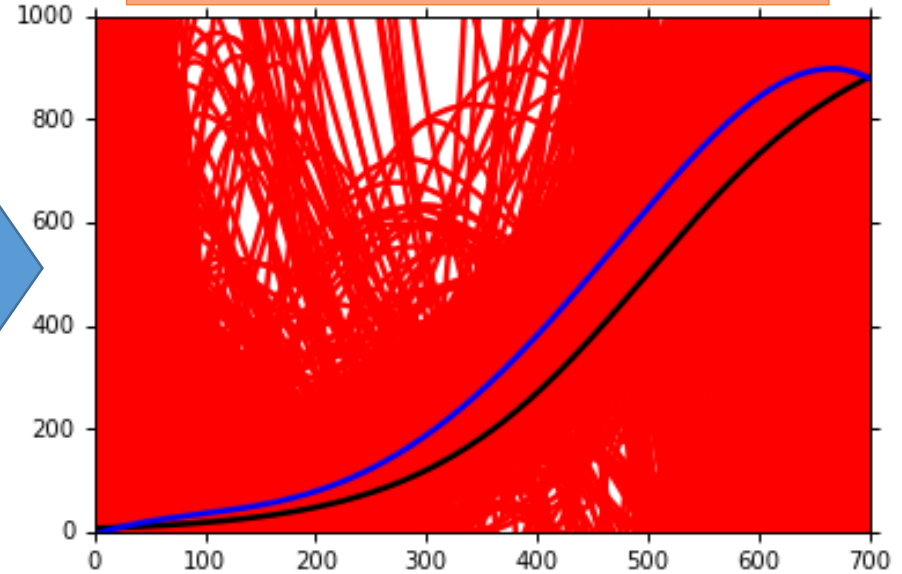


Bias

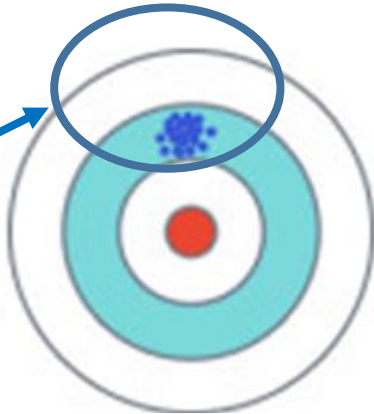
$$y = b + w \cdot x_{cp}$$



$$y = b + w_1 \cdot x_{cp} + w_2 \cdot (x_{cp})^2 + w_3 \cdot (x_{cp})^3 + w_4 \cdot (x_{cp})^4 + w_5 \cdot (x_{cp})^5$$

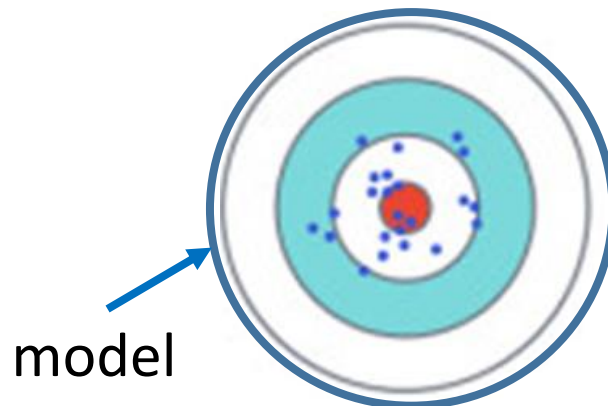


model



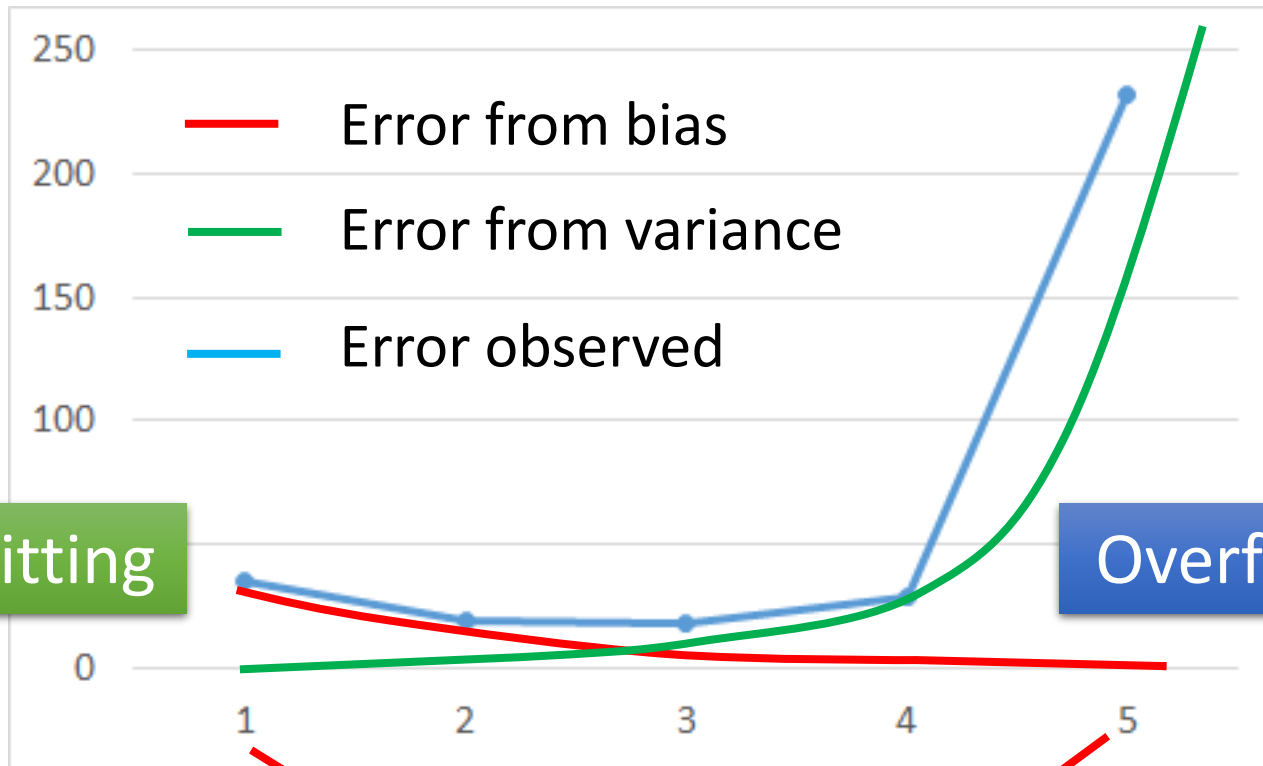
Large
Bias

model



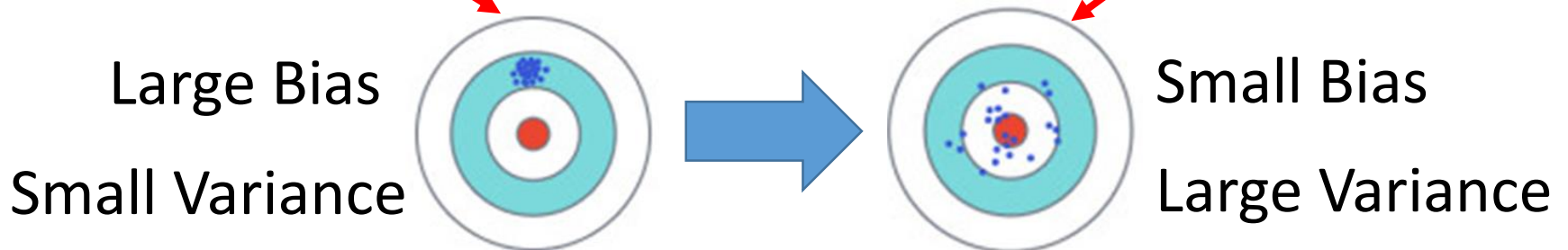
Small
Bias

Bias v.s. Variance



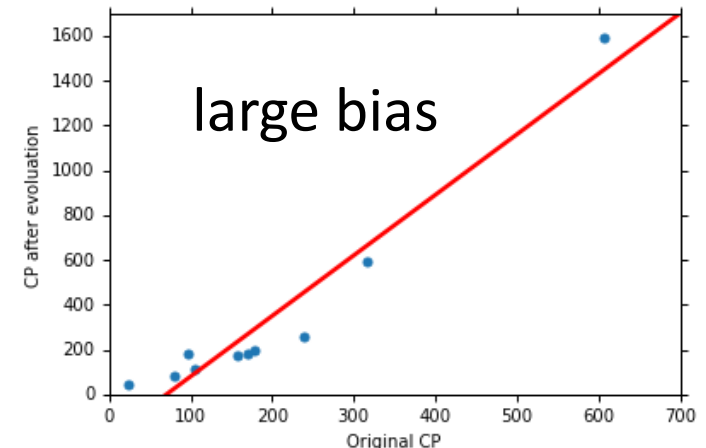
Underfitting

Overfitting



What to do with large bias?

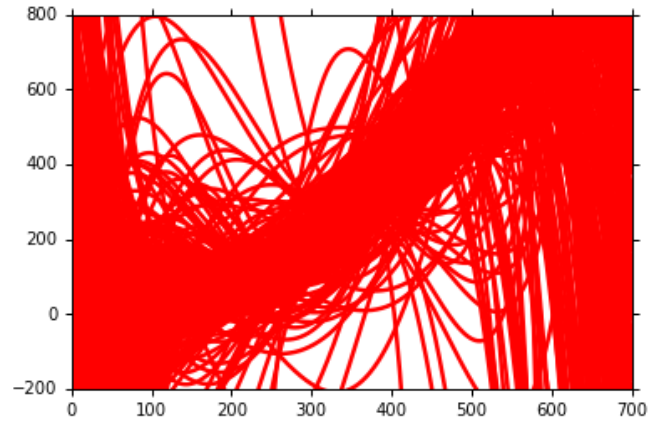
- Diagnosis:
 - If your model cannot even fit the training examples, then you have large bias **Underfitting**
 - If you can fit the training data, but large error on testing data, then you probably have large variance **Overfitting**
- For bias, redesign your model:
 - Add more features as input
 - A more complex model



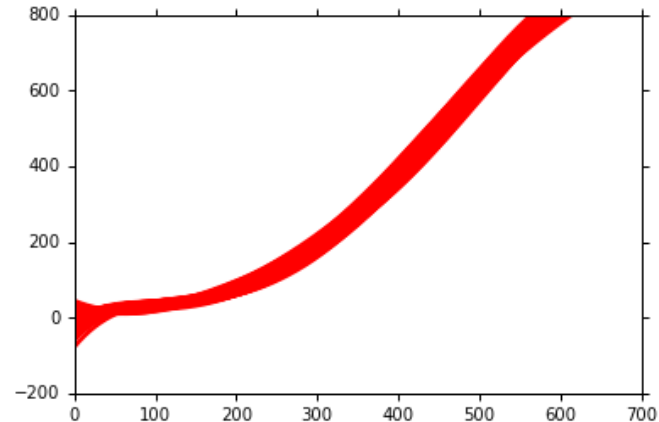
What to do with large variance?

- More data

Very effective,
but not always
practical

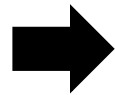


10 examples

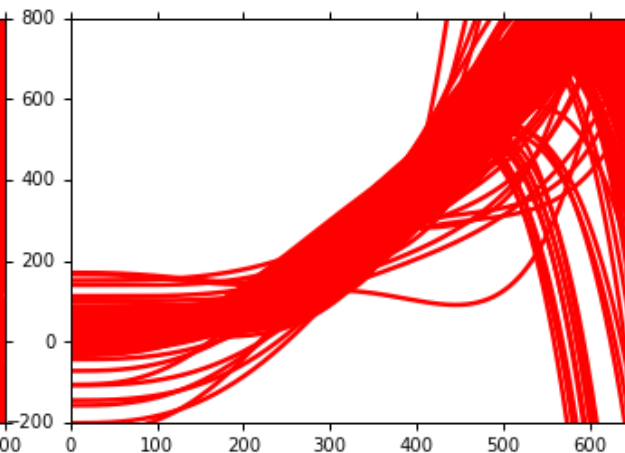
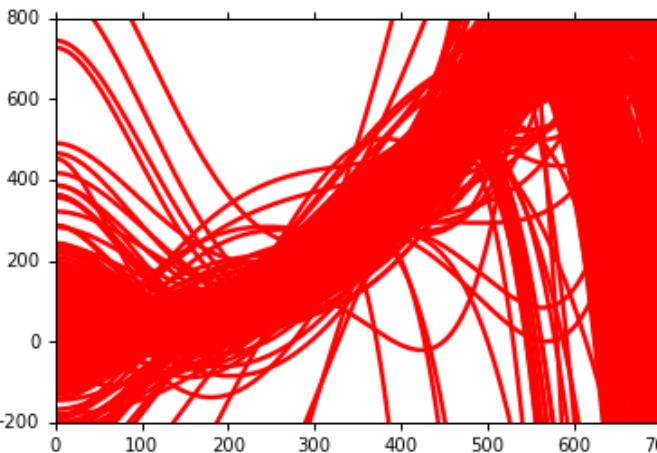
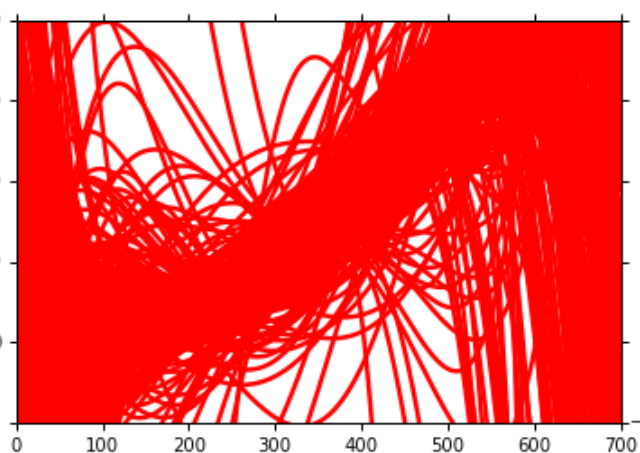


100 examples

- Regularization

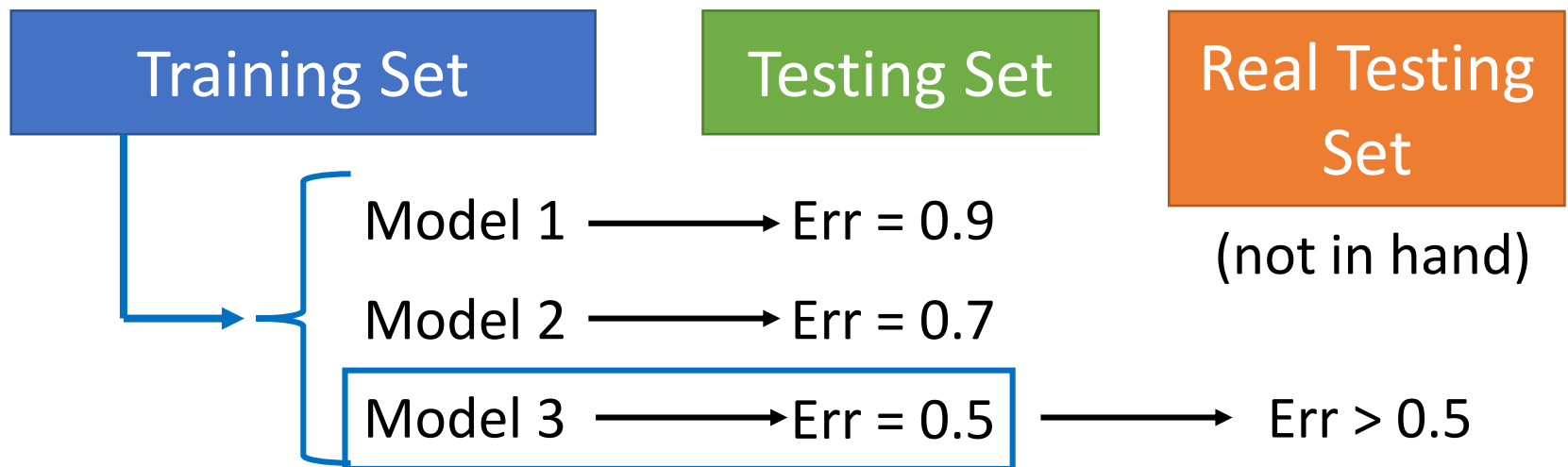


May increase bias



Model Selection

- There is usually a trade-off between bias and variance.
- Select a model that balances two kinds of error to minimize total error
- What you should NOT do:



Homework

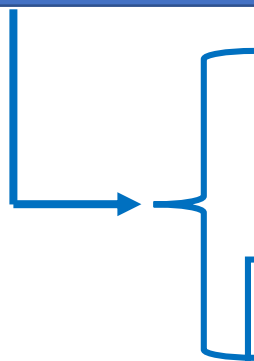
public

private

Training Set

Testing Set

Testing Set



Model 1 → Err = 0.9

Model 2 → Err = 0.7

Model 3 → Err = 0.5 → Err > 0.5

I beat baseline!

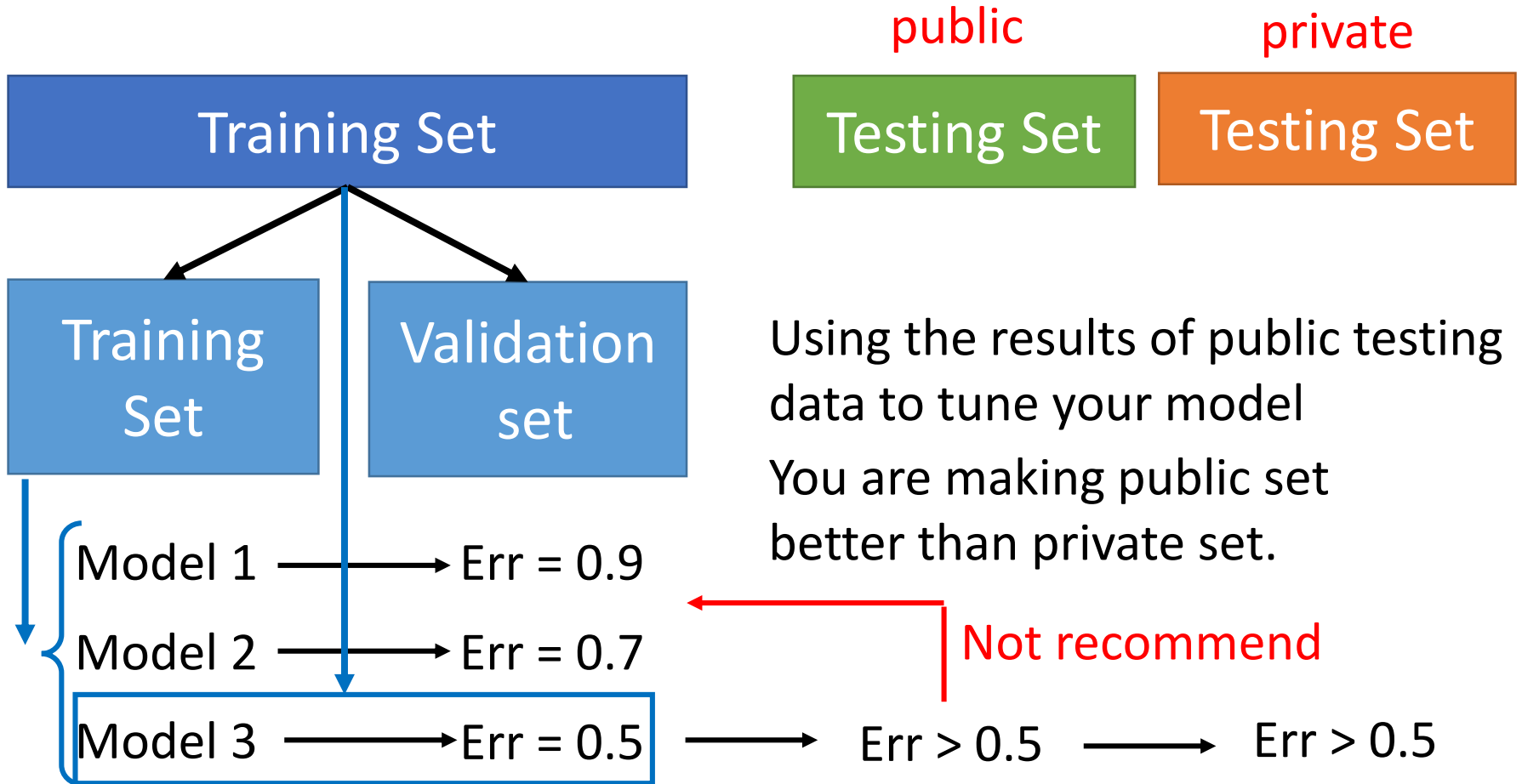
No, you don't

What will happen?

<http://www.chioka.in/how-to-select-your-final-models-in-a-kaggle-competitio/>



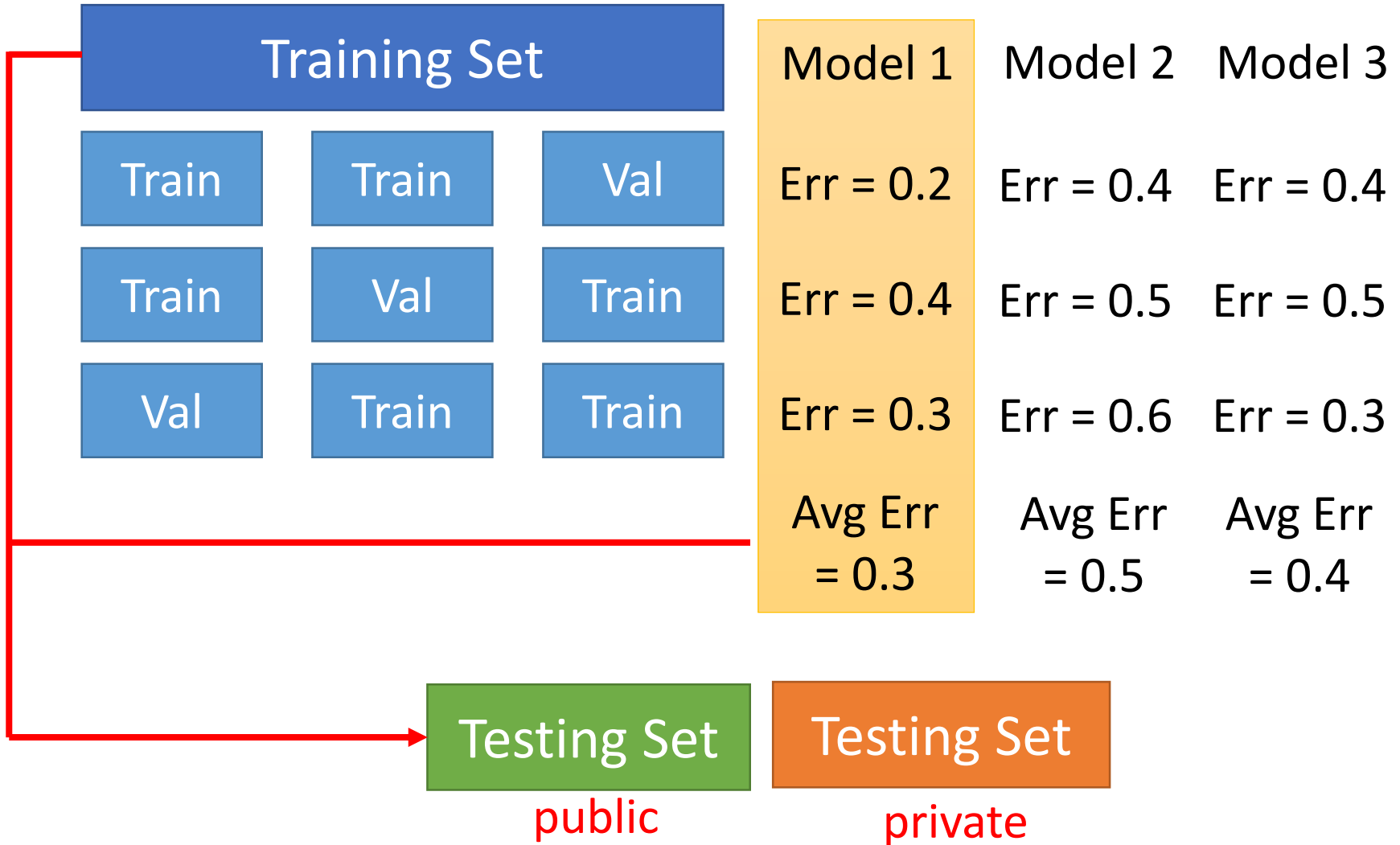
Model Selection Do and Don't



Testing data should never involve in model training nor model selection!!

K-fold Cross Validation

3-fold cross validation



Leave One Out (LOO) Cross Validation

Definition 1. Let H be a family of functions mapping from input space \mathcal{X} to output space \mathcal{Y} . Define the Leave One Out (LOO) cross validation error of algorithm $\mathcal{A} : \bigcup_{m \in \mathbb{N}} (\mathcal{X} \times \mathcal{Y})^m \rightarrow H$ on sample $S = ((x_i, y_i))_{i=1}^m \in (\mathcal{X} \times \mathcal{Y})^m$ as

$$\hat{\mathcal{R}}_S^{LOO}(\mathcal{A}) = \frac{1}{m} \sum_{i=1}^m \ell(h_{S_i}(x_i), y_i)$$

where $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_{\geq 0}$ is the loss function, $S_i = S \setminus \{(x_i, y_i)\}$, $h_{S_i} = \mathcal{A}(S_i)$.

Unbiased Estimation of Testing Error

Theorem 2. Let H be a family of functions mapping from input space \mathcal{X} to output space \mathcal{Y} , and let $\mathcal{A} : \bigcup_{m \in \mathbb{N}} (\mathcal{X} \times \mathcal{Y})^m \rightarrow H$. Let D be the unknown underlying distribution on $\mathcal{X} \times \mathcal{Y}$, then

$$\mathbb{E}_{S \sim D^m} [\hat{\mathcal{R}}_S^{LOO}(\mathcal{A})] = \mathbb{E}_{S' \sim D^{m-1}, (x,y) \in D} [\ell(\mathcal{A}(S')(x), y)]$$

In other words, LOO cross validation (on m instances) is an unbiased estimate of the algorithm's testing error (after training on $m - 1$ instances).

Proof. For $S = ((x_i, y_i))_{i=1}^m \in (\mathcal{X} \times \mathcal{Y})^m$, denote $S_i = S \setminus \{(x_i, y_i)\}$, $h_S = \mathcal{A}(S)$. Then

$$\begin{aligned} \mathbb{E}_{S \sim D^m} [\hat{\mathcal{R}}_S^{LOO}(\mathcal{A})] &= \frac{1}{m} \sum_{i=1}^m \mathbb{E}_{S \sim D^m} [\ell(h_{S_i}(x_i), y_i)] \\ &= \frac{1}{m} \sum_{i=1}^m \mathbb{E}_{S_i \sim D^{m-1}, (x_i, y_i) \sim D} [\ell(h_{S_i}(x_i), y_i)] \\ &= \mathbb{E}_{S' \sim D^{m-1}, (x,y) \in D} [\ell(h_{S'}(x), y)]. \end{aligned}$$

□