The Mathematics of Machine Learning 2026

Lecture 2

Tim van Erven

University of Amsterdam

Outline

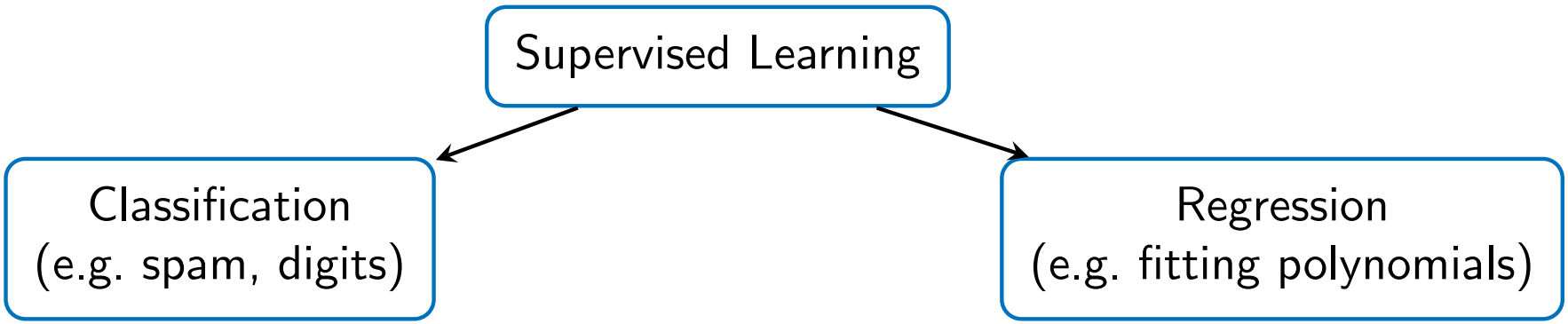
1. Organization
2. Warm-up
3. Curse of Dimensionality
4. Statistical Decision Theory
5. Empirical Risk Minimization
6. Cross-validation

Organization

Exercise Classes on Thursday:

- Look up your group (A or B) on Canvas
- Bring a laptop!

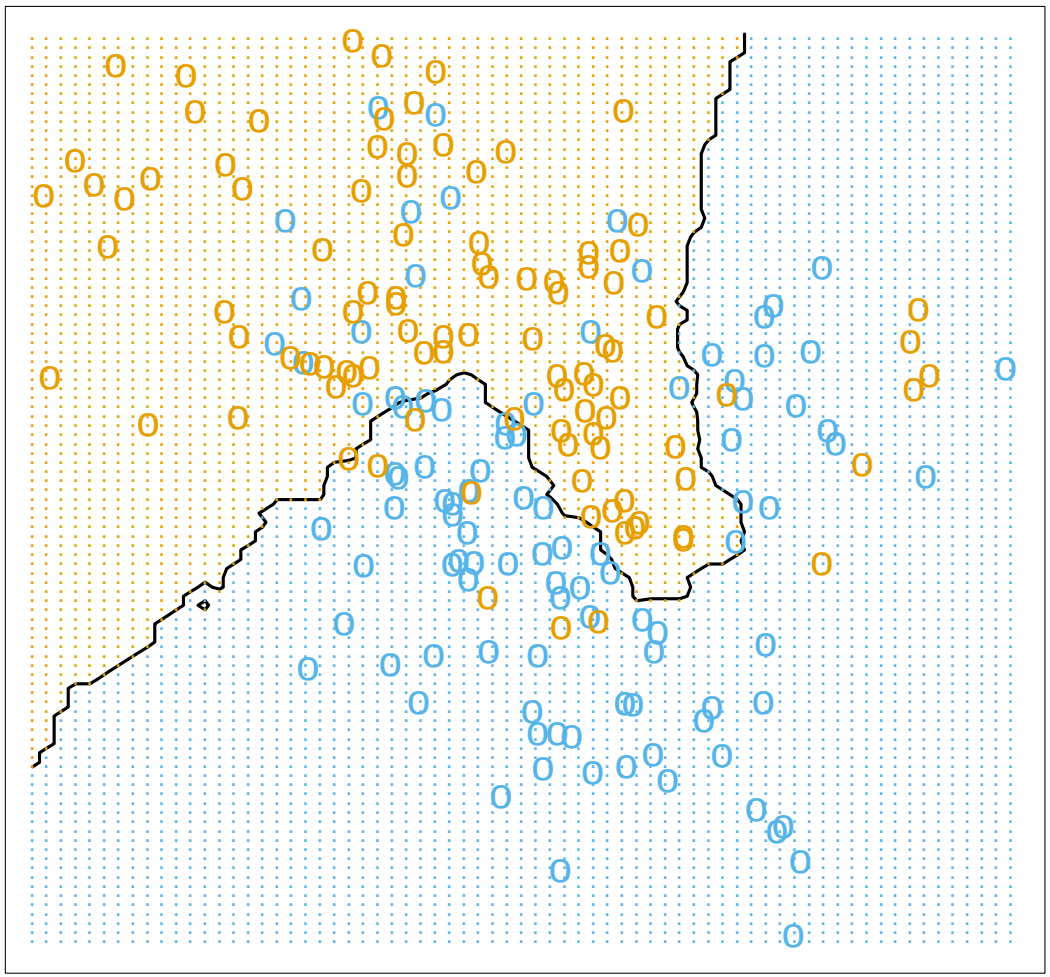
Warm-up



$$T = \begin{pmatrix} Y_1 \\ X_1 \end{pmatrix}, \ldots, \begin{pmatrix} Y_N \\ X_N \end{pmatrix}$$

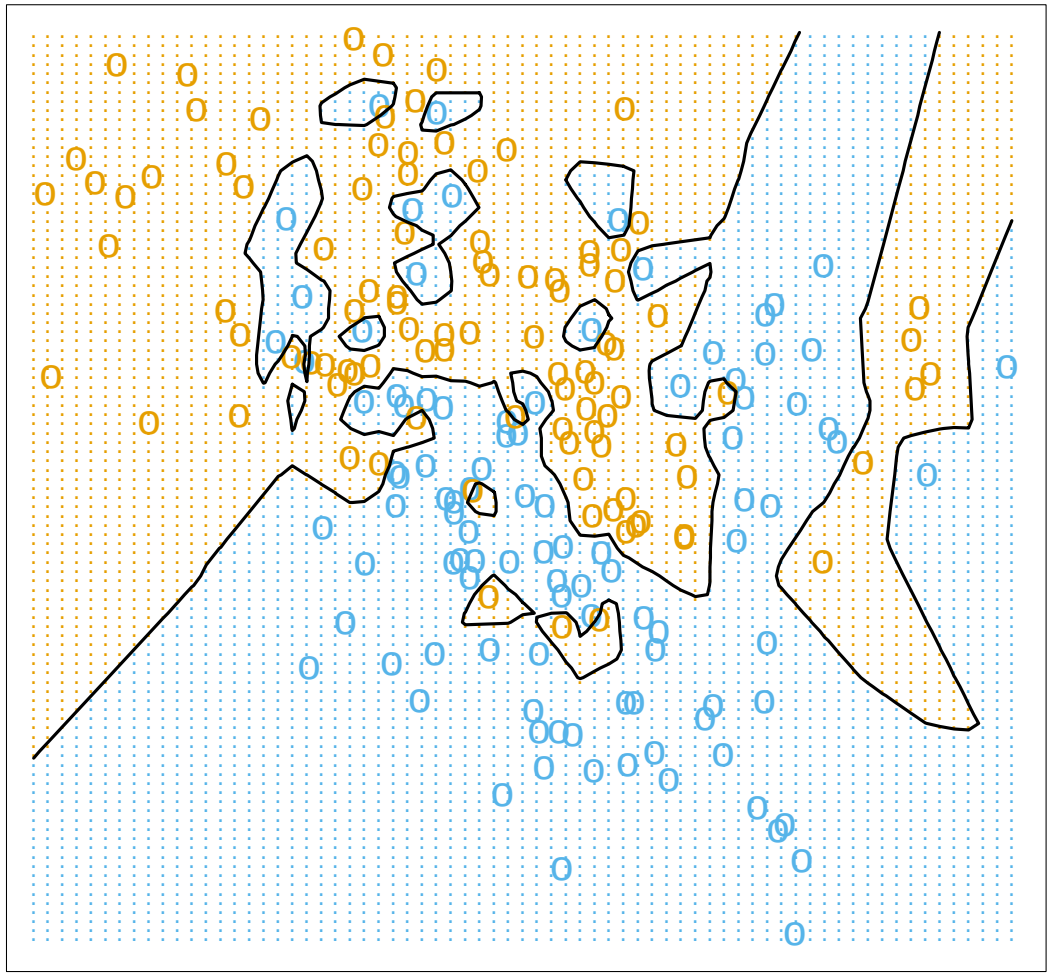
$\longrightarrow \hat{f} \in \mathcal{F}$ and predict $\hat{Y} = \hat{f}(X)$ for new X

K-Nearest Neighbor Classification



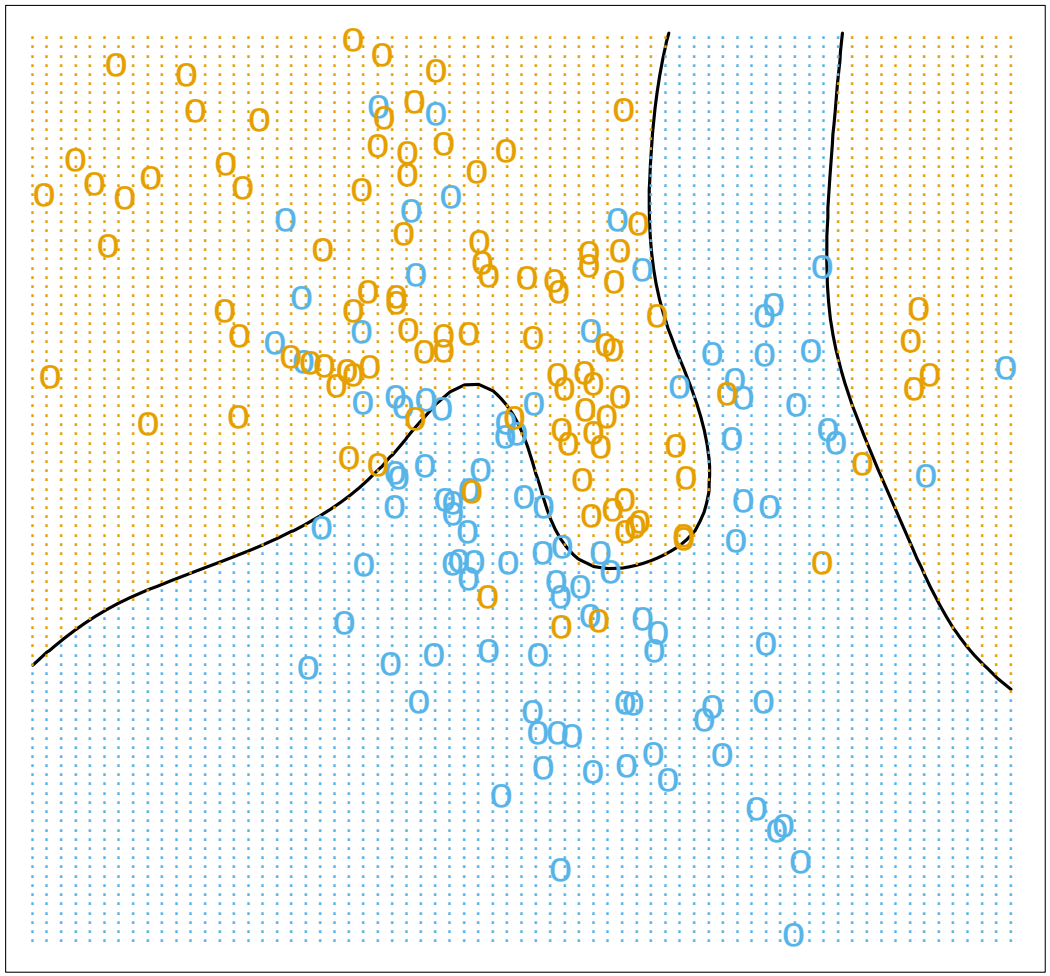
K-Nearest Neighbor Classification

1-Nearest Neighbor Classifier

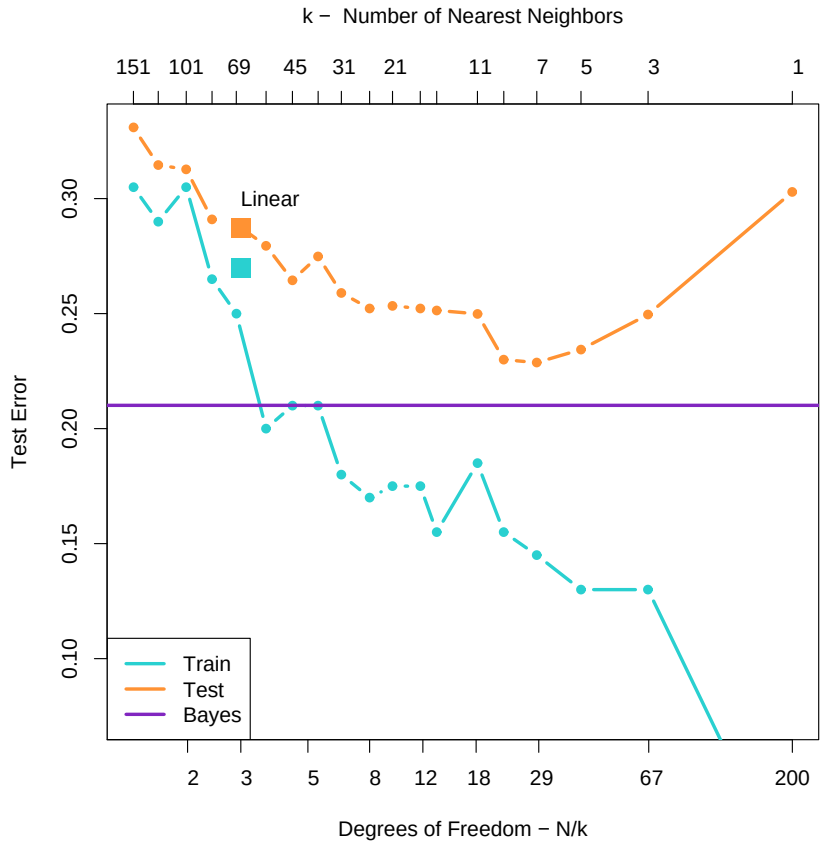


K-Nearest Neighbor Classification

Bayes Optimal Classifier

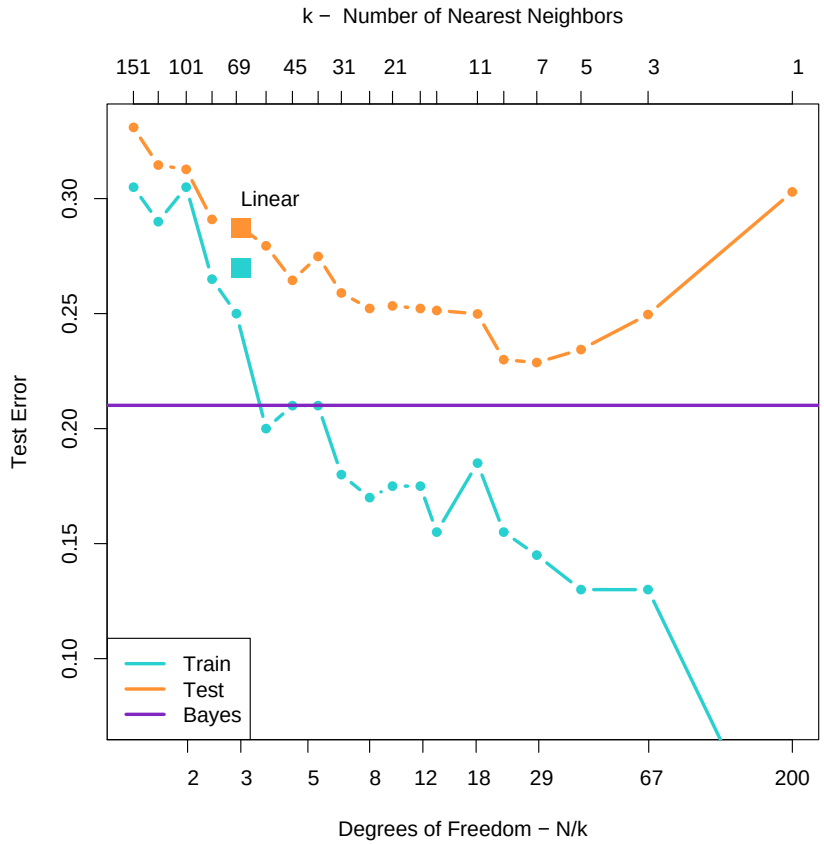


K-Nearest Neighbor Classification



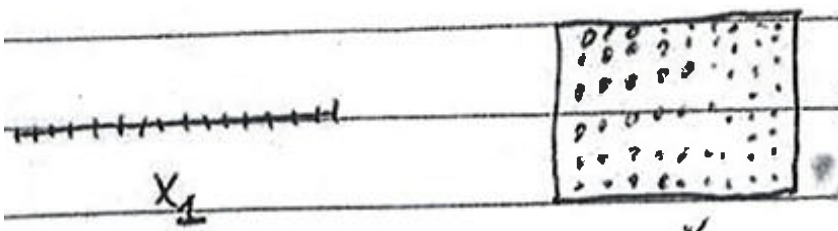
- Nearest neighbor always better than linear model?
  $\rightarrow$ **Curse of dimensionality!**

K-Nearest Neighbor Classification

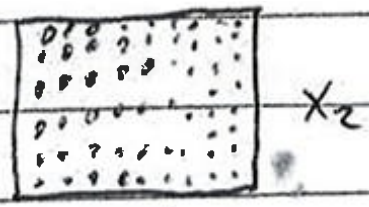


- Nearest neighbor always better than linear model?
  $\rightarrow$ **Curse of dimensionality!**
- **Optimal classifier** $\rightarrow$ **Statistical decision theory**

Curse of Dimensionality



1 feature
N samples

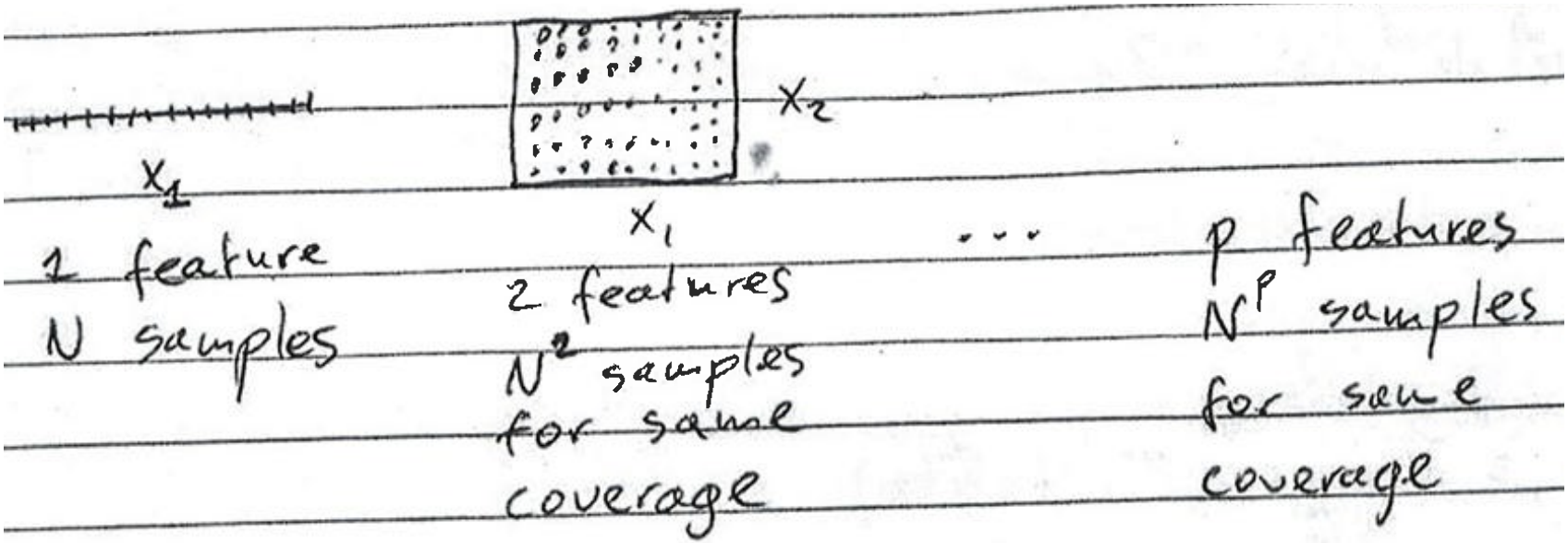


2 features
N^2 samples
for same
coverage

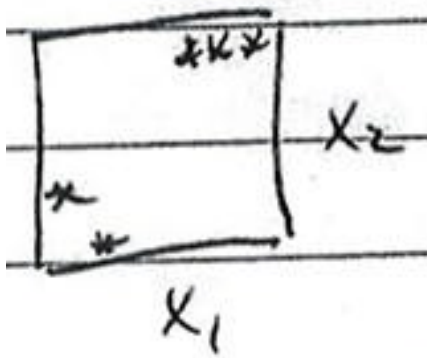
...

p features
N^p samples
for same
coverage

Curse of Dimensionality



Conclusion: In high dimensions you only see samples from a very small part of the domain:



"A high-dimensional space is a lonely place." – Bernard Schölkopf

Statistical Decision Theory: Loss Function

- Learn $\hat{f}$ from training data $T = \begin{pmatrix} Y_1 \\ X_1 \end{pmatrix}, \ldots, \begin{pmatrix} Y_N \\ X_N \end{pmatrix} \overset{\text{i.i.d.}}{\sim} P^*$
- Evaluate on new $\begin{pmatrix} Y \\ X \end{pmatrix} \sim P^*$

Loss function:

- $L(Y, \hat{Y})$ measures how bad it is if truth is Y, but we predict $\hat{Y} = \hat{f}(X)$ (smaller is better)
- Examples:
  - Regression: $L(Y, \hat{Y}) = (Y - \hat{Y})^2$ **"squared error"**
  - Classification: $L(Y, \hat{Y}) = \begin{cases} 0 & \text{if } \hat{Y} = Y \\ 1 & \text{if } \hat{Y} \neq Y \end{cases}$ **"0/1-loss"**
  - Other choices possible!
    (Depends on what is important in your application.)

---

i.i.d. = independent, identically distributed

Statistical Decision Theory: EPE

often also called "risk"

$\begin{pmatrix} Y \\ X \end{pmatrix} \sim P^*$ random, so want small **expected prediction error**:

$$\mathrm{EPE}(f) = \mathop{\mathbb{E}}_{(X,Y)} \Big[L\big(Y, f(X)\big) \Big]$$

Statistical Decision Theory: EPE

often also called "risk"

$\begin{pmatrix} Y \\ X \end{pmatrix} \sim P^*$ random, so want small **expected prediction error**:

$$\text{EPE}(f) = \mathop{\mathbb{E}}_{(X,Y)} \Big[L\big(Y, f(X)\big) \Big]$$

- We want $\hat{f}$ such that $\mathbb{E}_{\mathcal{T}}[\text{EPE}(\hat{f})]$ as small as possible.

Statistical Decision Theory: EPE

often also called "risk"

$\begin{pmatrix} Y \\ X \end{pmatrix} \sim P^*$ random, so want small **expected prediction error**:

$$\text{EPE}(f) = \mathop{\mathbb{E}}_{(X,Y)} \Big[L\big(Y, f(X)\big) \Big]$$

- We want $\hat{f}$ such that $\mathbb{E}_T[\text{EPE}(\hat{f})]$ as small as possible.

Classification:
$$\begin{aligned} \text{EPE}(f) &= \Pr(f(X) \neq Y) \cdot 1 + \Pr(f(X) = Y) \cdot 0 \\ &= \Pr(f(X) \neq Y) \end{aligned}$$

Regression:
$$\text{EPE}(f) = \mathbb{E}\Big[(Y - f(X))^2 \Big]$$

Statistical Decision Theory: Bayes-optimal Predictor

Bayes-optimal f_B **minimizes** EPE(f) over all possible prediction functions f:

$$f_B = \arg\min_f \mathbb{E}\left[L(Y, f(X))\right]$$

$$f_B(X) = \arg\min_f \mathbb{E}\left[L(Y, f(X)) \mid X\right]$$

Statistical Decision Theory: Bayes-optimal Predictor

Bayes-optimal f_B **minimizes** EPE(f) over all possible prediction functions f:

$$f_B = \arg\min_f \mathbb{E}\left[L(Y, f(X))\right]$$

$$f_B(X) = \arg\min_f \mathbb{E}\left[L(Y, f(X)) \mid X\right]$$

- **Optimal** = no learning algorithm can do better by definition
- **Unknown**, because depends on P^*
- Can still make mistakes because of **unpredictability** of Y given X
- (Nothing to do with Bayesian statistics)

Statistical Decision Theory: Bayes-optimal Predictor

Bayes-optimal f_B **minimizes** EPE(f) over all possible prediction functions f:

$$f_B = \arg\min_f \mathbb{E}\left[L(Y, f(X))\right]$$

$$f_B(X) = \arg\min_f \mathbb{E}\left[L(Y, f(X)) \mid X\right]$$

Classification with 0/1-loss:

$$\begin{aligned} f_B(X) &= \arg\min_{\hat{Y}} \Pr(\hat{Y} \neq Y \mid X) \\ &= \arg\max_{\hat{Y}} \Pr(\hat{Y} = Y \mid X) \\ &= \begin{cases} 1 & \text{if } \Pr(Y = 1|X) \geq 0.5 \\ 0 & \text{if } \Pr(Y = 1|X) < 0.5 \end{cases} \end{aligned}$$

(for two classes)

Statistical Decision Theory: Bayes-optimal Predictor

Bayes-optimal f_B **minimizes** EPE(f) over all possible prediction functions f:

$$f_B = \arg\min_f \mathbb{E}\left[L(Y, f(X))\right]$$

$$f_B(X) = \arg\min_f \mathbb{E}\left[L(Y, f(X)) \mid X\right]$$

Regression with squared error:

$$f_B(X) = \arg\min_{\hat{Y}} \mathbb{E}[(Y - \hat{Y})^2 \mid X]$$

$$= \mathbb{E}[Y \mid X]$$

Statistical Decision Theory: Bayes-optimal Predictor

Bayes-optimal f_B **minimizes** EPE(f) over all possible prediction functions f:

$$f_B = \arg\min_f \mathbb{E}\left[L(Y, f(X))\right]$$

$$f_B(X) = \arg\min_f \mathbb{E}\left[L(Y, f(X)) \mid X\right]$$

Regression with squared error:

$$f_B(X) = \arg\min_{\hat{Y}} \mathbb{E}[(Y - \hat{Y})^2 \mid X]$$

$$= \mathbb{E}[Y \mid X]$$

If $Y = f^*(X) + \text{noise}$ for $\mathbb{E}[\text{noise} \mid X] = 0$, then
$f_B(X) = \mathbb{E}[Y \mid X] = \mathbb{E}[f^*(X) + \text{noise} \mid X] = f^*(X)$

Summary

$\text{EPE}(f) = \mathbb{E}_{(X,Y)}\left[L\big(Y, f(X)\big)\right]$ measures quality of f

- Regression: $L\big(Y, f(X)\big) = \big(Y - f(X)\big)^2$ "squared error"
- Classification: $L(Y, \hat{Y}) = \begin{cases} 0 & \text{if } \hat{Y} = Y \\ 1 & \text{if } \hat{Y} \neq Y \end{cases}$ "0/1-loss"

Summary

$\mathrm{EPE}(f) = \mathbb{E}_{(X,Y)}\left[L\big(Y, f(X)\big)\right]$ measures quality of f

- Regression: $L\big(Y, f(X)\big) = \big(Y - f(X)\big)^2$ "squared error"
- Classification: $L(Y, \hat{Y}) = \begin{cases} 0 & \text{if } \hat{Y} = Y \\ 1 & \text{if } \hat{Y} \neq Y \end{cases}$ "0/1-loss"

Bayes-optimal predictor: $f_{\mathrm{B}} = \arg\min_f \mathrm{EPE}(f)$

- Regression: $f_{\mathrm{B}}(X) = \mathbb{E}[Y \mid X]$
- Classification: $f_{\mathrm{B}}(X) = \arg\max_{\hat{Y}} \Pr(\hat{Y} = Y \mid X)$

Summary

$\mathrm{EPE}(f) = \mathbb{E}_{(X,Y)}\left[L\big(Y, f(X)\big)\right]$ measures quality of f

- Regression: $L\big(Y, f(X)\big) = \big(Y - f(X)\big)^2$ "squared error"
- Classification: $L(Y, \hat{Y}) = \begin{cases} 0 & \text{if } \hat{Y} = Y \\ 1 & \text{if } \hat{Y} \neq Y \end{cases}$ "0/1-loss"

Bayes-optimal predictor: $f_{\mathrm{B}} = \arg\min_f \mathrm{EPE}(f)$

- Regression: $f_{\mathrm{B}}(X) = \mathbb{E}[Y \mid X]$
- Classification: $f_{\mathrm{B}}(X) = \arg\max_{\hat{Y}} \Pr(\hat{Y} = Y \mid X)$

Estimators:

$\hat{f}$ depends on $T \rightarrow \mathrm{EPE}(\hat{f})$ depends on $T \rightarrow$ evaluate $\mathbb{E}_T[\mathrm{EPE}(\hat{f})]$

Empirical Risk Minimization (ERM)

- Want to minimize EPE(f), but don't know P^*
- **Approximate** P^* by empirical distribution of training data T, and
- Choose $f \in \mathcal{F}$ that minimizes

$$\frac{1}{N} \sum_{i=1}^{N} L(Y_i, f(X_i))$$

Classification:

- Minimize nr. of mistakes on training data
- Usually cannot compute efficiently for 0/1-loss (unless zero mistakes is achievable on T)

Regression:

- Minimize sum of squared errors = "least squares"
- Can compute (by formula) if $\mathcal{F}$ is linear model

Empirical Risk Minimization (ERM)

- Want to minimize EPE(f), but don't know P^*
- **Approximate** P^* by empirical distribution of training data T, and
- Choose $f \in \mathcal{F}$ that minimizes

$$\frac{1}{N} \sum_{i=1}^{N} L(Y_i, f(X_i))$$

Is this always best? No!

1. Potential **overfitting** (see K-NN picture train and test error as function of K)
2. Restricted to model $\mathcal{F}$ (potential **underfitting**)

K-Fold Cross-Validation: Motivation

Hyperparameter $m \in \{1, \ldots, M\}$ that controls model complexity. E.g.

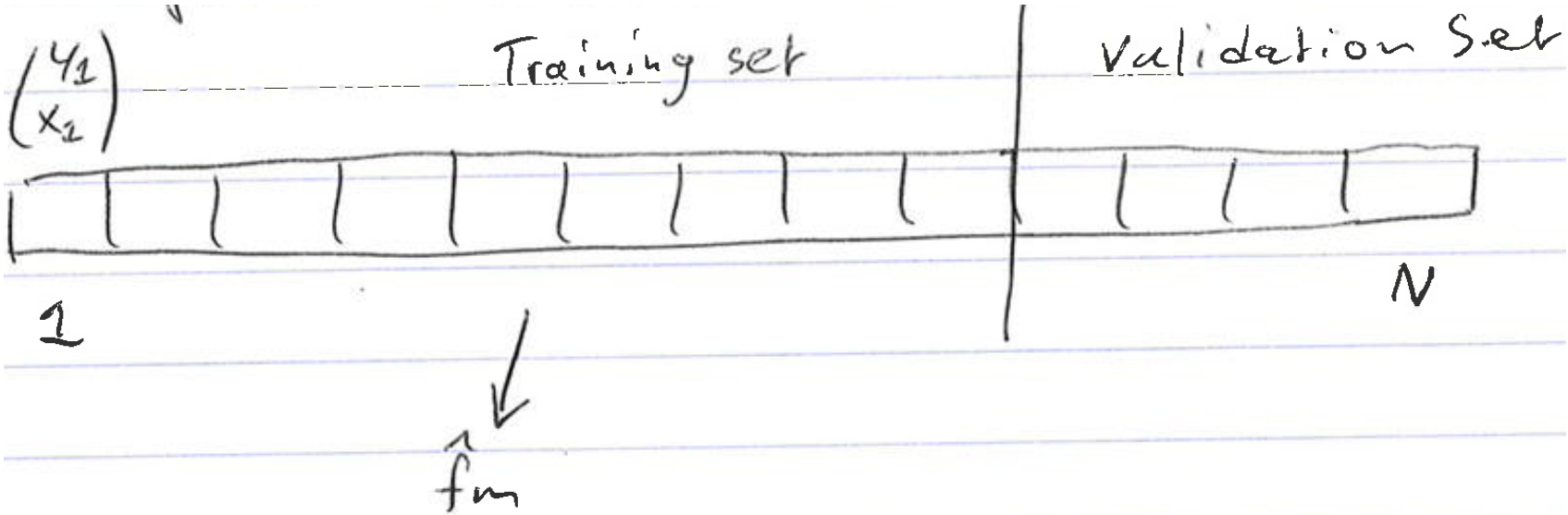
- m is degree of polynomial in linear regression
- m is K in K-nearest neighbor

Training set $T \to \hat{f}_m$ for each m

Q. What happens if we look at fit on training data to choose m (i.e. we use ERM)

- for degree of polynomial?
- for K in K-NN?

Hold-out Estimate



Hold-out estimate:

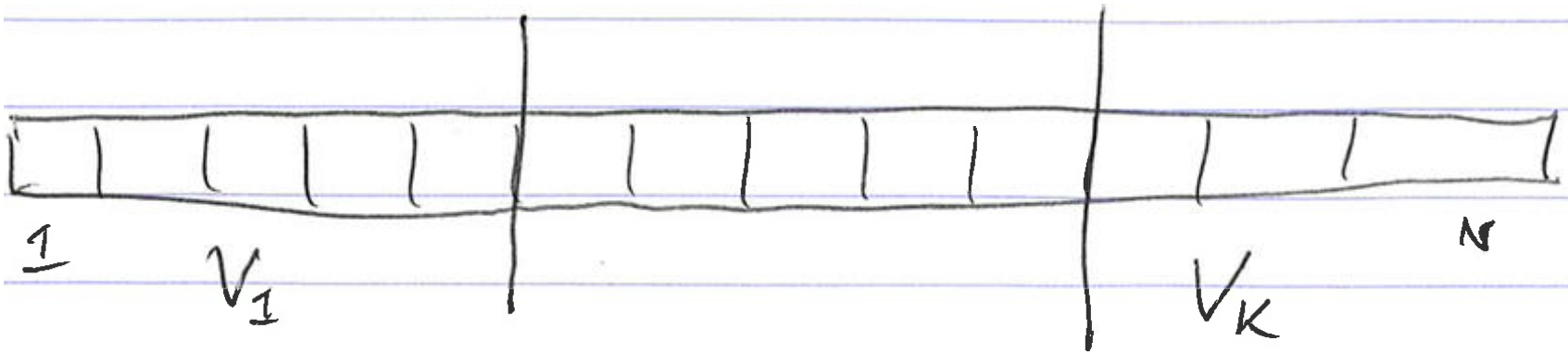
- Randomly set aside validation set V
- Construct $\hat{f}_m$ on rest of training data for all m
- Use ERM on validation set:

$$\hat{m} = \arg\min_m \frac{1}{|V|} \sum_{i \in V} L\big(Y_i, \hat{f}_m(X_i)\big)$$

- Works well if V is large enough, but that reduces training set a lot

K-Fold Cross Validation

Try to get away with smaller validation set by averaging multiple choices:



For $k = 1, \ldots, K$:

- Train $\hat{f}_m^k$ on all data except V_k (for each m)
- Evaluate on V_k: $\hat{L}_m^k = \frac{1}{|V_k|} \sum_{i \in V_k} L(Y_i, \hat{f}_m(X_i))$

Average over all choices of validation set: $\hat{m} = \arg\min_m \frac{1}{K} \sum_{k=1}^{K} \hat{L}_m^k$

K-Fold Cross Validation

For $k = 1, \ldots, K$:

- Train $\hat{f}_m^k$ on all data except V_k (for each m)
- Evaluate on V_k: $\hat{L}_m^k = \frac{1}{|V_k|} \sum_{i \in V_k} L\big(Y_i, \hat{f}_m(X_i)\big)$

Average over all choices of validation set: $\hat{m} = \arg\min_m \frac{1}{K} \sum_{k=1}^{K} \hat{L}_m^k$

For $\hat{m}$, we now have K predictors $\hat{f}_{\hat{m}}^1, \ldots, \hat{f}_{\hat{m}}^K$.
How do we get a **single one?**

1. **In practice:** Throw away predictors, and train final $\hat{f}_{\hat{m}}$ on all data
2. **In theory:** combine K predictors. E.g. $\hat{f}_{\hat{m}}(X) = \frac{1}{K} \sum_{k=1}^{K} \hat{f}_{\hat{m}}^k(X)$

K-Fold Cross Validation

For $k = 1, \ldots, K$:

- Train $\hat{f}_m^k$ on all data except V_k (for each m)
- Evaluate on V_k: $\hat{L}_m^k = \frac{1}{|V_k|} \sum_{i \in V_k} L\big(Y_i, \hat{f}_m(X_i)\big)$

Average over all choices of validation set: $\hat{m} = \arg\min_m \frac{1}{K} \sum_{k=1}^{K} \hat{L}_m^k$

$K = N$: "leave-one-out cross-validation"[1]

Best K is complicated [Arlot and Celisse, 2010]. Depends on:

- Bias: estimates $\hat{f}_m$ trained on $N - N/K$ data points. Might have obtained better estimator if trained on all data
  $\rightarrow$ small K might be pessimistic: overestimate EPE
- Variance as we increase K:
  - Smaller validation set size $N/K \rightarrow$ increases variance
  - But more folds decreases variances
  - Stability of estimators $\hat{f}_m$ also has effect

[1] (Mervyn Stone anecdote)

K-Fold Cross Validation

For $k = 1, \ldots, K$:

- Train $\hat{f}_m^k$ on all data except V_k (for each m)
- Evaluate on V_k: $\hat{L}_m^k = \frac{1}{|V_k|} \sum_{i \in V_k} L\big(Y_i, \hat{f}_m(X_i)\big)$

Average over all choices of validation set: $\hat{m} = \arg\min_m \frac{1}{K} \sum_{k=1}^{K} \hat{L}_m^k$

Best K is complicated [Arlot and Celisse, 2010]. Depends on:

- Bias: estimates $\hat{f}_m$ trained on $N - N/K$ data points. Might have obtained better estimator if trained on all data
  $\rightarrow$ small K might be pessimistic: overestimate EPE
- Variance as we increase K:
  - Smaller validation set size $N/K \rightarrow$ increases variance
  - But more folds decreases variances
  - Stability of estimators $\hat{f}_m$ also has effect

K-Fold Cross Validation

For $k = 1, \ldots, K$:

- Train $\hat{f}_m^k$ on all data except V_k (for each m)
- Evaluate on V_k: $\hat{L}_m^k = \frac{1}{|V_k|} \sum_{i \in V_k} L\big(Y_i, \hat{f}_m(X_i)\big)$

Average over all choices of validation set: $\hat{m} = \arg\min_m \frac{1}{K} \sum_{k=1}^{K} \hat{L}_m^k$

Best K is complicated [Arlot and Celisse, 2010]. Depends on:

- Bias: estimates $\hat{f}_m$ trained on $N - N/K$ data points. Might have obtained better estimator if trained on all data
  $\rightarrow$ small K might be pessimistic: overestimate EPE
- Variance as we increase K:
  - Smaller validation set size $N/K \rightarrow$ increases variance
  - But more folds decreases variances
  - Stability of estimators $\hat{f}_m$ also has effect

In practice: $K = 5$ **or** $K = 10$ is good in many cases (see book for references)

References

S. Arlot and A. Celisse. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40–79, 2010. doi: 10.1214/09-SS054. URL https://doi.org/10.1214/09-SS054.

Extra slide: Bias-Variance Decomposition for Squared Error

Understand balance between underfitting and overfitting in regression:

Theorem (Bias-Variance Decomposition)

Let $\bar{f}(X) = \mathbb{E}_T[\hat{f}(X)]$. *Then*

$$\begin{aligned}
\mathop{\mathbb{E}}_{T}[\mathrm{EPE}(\hat{f})] = \mathop{\mathbb{E}}_{X}[\mathrm{Var}(Y \mid X)] & \qquad \leftarrow \mathrm{EPE}(f_{\mathrm{B}}) \\
+ \mathop{\mathbb{E}}_{X}\left[\left(f_{\mathrm{B}}(X) - \bar{f}(X)\right)^2\right] & \qquad \leftarrow \textit{bias} \\
+ \mathop{\mathbb{E}}_{T}\mathop{\mathbb{E}}_{(X,Y)}\left[\left(\hat{f}(X) - \bar{f}(X)\right)^2\right] & \qquad \leftarrow \textit{variance}
\end{aligned}$$

- Bias typically smaller for larger $\mathcal{F}$
- Variance typically larger for larger $\mathcal{F}$