香港城市大學
City University of Hong Kong
專業 創新 胸懷全球
Professional · Creative
For The World

Evaluation:
The Problem of Overfitting

CS5483 Data Warehousing and Data Mining

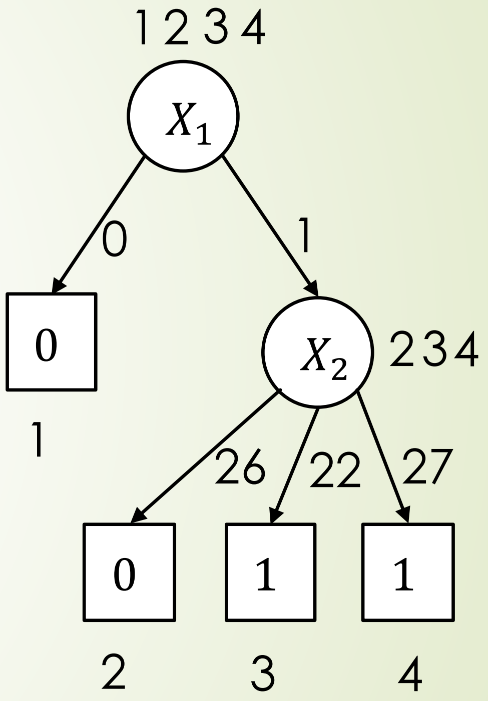
Guess the value of y

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1
	1	22	y

- $y =$ __ because ____________.

Guess the value of y for any (x_1, x_2)

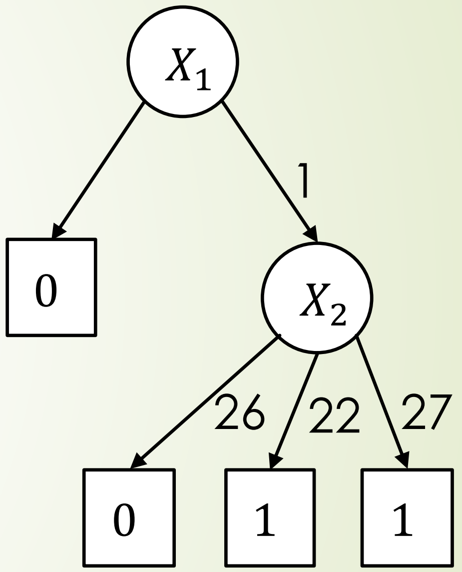
	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1
	x_1	x_2	y



- **D**____________ **t**_____.
- Is it making good decisions? It **f**___ the data **p**____________.
- Is it simple? _______

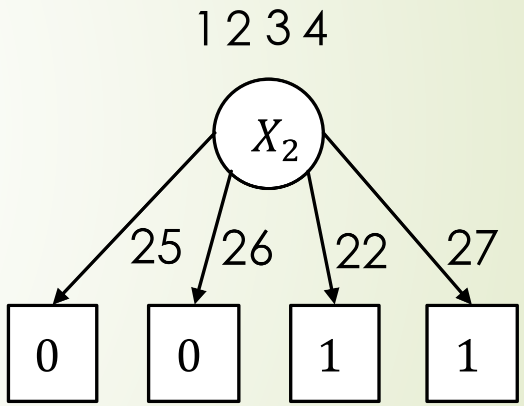
Guess the value of y for any (x_1, x_2)

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1
	x_1	x_2	y



Guess the value of y for any (x_1, x_2)

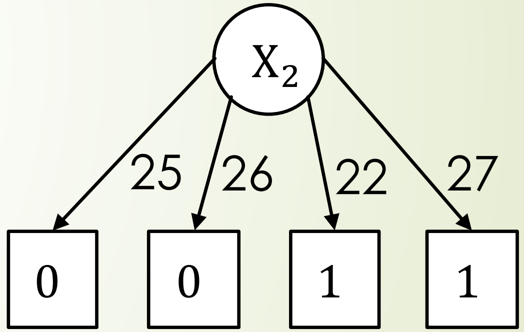
	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1
	x_1	x_2	y



- Can have a smaller tree because X_2 completely ________________ Y.
- Smaller tree means faster computation and lower storage.
- Any other benefit? What if $(x_1, x_2) = (0,22)$? $y =$_____ (generalize to u______ data.)

Does the classifier generalize well?

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1
	x_1	x_2	y



- Y: Diagnosis of certain disease.
- X_1: Result of a medical test.
- X_2: Temperature at which the test is conducted.
- If $(x_1, x_2) = (0,22)$, should $y = 1$? Yes/No, because ________________________________.

Overfitting

- Learning patterns that
  - **fits** the data well, but
  - not **g**_______________ well to new cases (future data).
- The challenges:
  1. How to estimate the actual performance on unseen data, not the **overly-optimistic** performance on fitted data?
  2. How to learn the desired knowledge, i.e., patterns that **generalize** well to unseen data?
- What causes overfitting?

Naive Formulation

$\mathbf{X} \coloneqq (\mathrm{X}_1, \mathrm{X}_2, \dots, \mathrm{X}_m)$

Y

Equal?

$\hat{f}(\mathbf{X})$

$\hat{f}: \mathcal{X} \to \mathcal{Y}$

- **Input feature** vector $\mathbf{X}$ from **feature space** $\mathcal{X}$.
- **Target** Y from a discrete set $\mathcal{Y}$:
  - B__________ classification:$|\mathcal{Y}| = 2$
  - M__________ classification:$|\mathcal{Y}| > 2$
- Obtain a **classifier** $\hat{f}: \mathcal{X} \to \mathcal{Y}$ to predict Y from $\mathbf{X}$.
- What is a good classifier?

Naive Formulation

$\mathbf{X} \coloneqq (X_1, X_2, \ldots, X_m)$ $\qquad$ Y

$\hat{f}: \mathcal{X} \to \mathcal{Y}$ $\qquad$ $\hat{f}(\mathbf{X})$

$$\hat{f} \in \arg\min_{f:\mathcal{X}\to\mathcal{Y}} P[Y \neq f(\mathbf{X})]$$

is a **classifier** that

- Minimize $\Pr[Y \neq f(\mathbf{X})]$, **e**______ probability.
- Maximize $\Pr[Y = f(\mathbf{X})]$, **a**________________.

- More generally, choose $\hat{f} = f_w$ where w is a solution to

$$\min_{w \in \mathcal{W}} \overbrace{E\left[\underbrace{\mathbb{1}(Y \neq f_w(\mathbf{X}))}_{}\right]}^{R(w)}$$

$$\min_{w\in\mathcal{W}} E\Big[\overbrace{\mathbb{1}\big(Y \neq f_w(\mathbf{X})\big)}^{L(Y, f_w(\mathbf{X}))}\Big]$$

  - $L: \mathcal{Y}^2 \to \mathbb{R}$ can be any **l**___ function such as the 0-1 loss $(y, \hat{y}) \mapsto \mathbb{1}(y \neq \hat{y})$.
  - $R: \mathcal{W} \to \mathbb{R}$ is the corresponding **r**_____ functional over a (compact) hypothesis space $\mathcal{W}$ for any class $\{f_w | w \in \mathcal{W}\}$ of functions.

Naive Formulation

$\mathbf{X} \coloneqq (\mathrm{X}_1, \mathrm{X}_2, \dots, \mathrm{X}_m)$ $\quad$ Y

$w \in \mathcal{W}$ $\quad$ $f_w(\mathbf{X})$

$$\min_{w \in \mathcal{W}} \overbrace{\int_{(x,y)\in\mathcal{X}\times\mathcal{Y}} L\big(y, f_w(x)\big)\, dP_{\mathbf{XY}}(x,y)}^{R(w)}$$

- Optimal? Yes/No because ______________________________________.
- Cannot use in practice because the **d**__________ is often unknown.

Empirical Risk Minimization (ERM)

$\mathbf{X} \coloneqq (\mathrm{X}_1, \mathrm{X}_2, \dots, \mathrm{X}_m)$ $\quad\quad$ Y

$\hat{f}: \mathcal{X} \to \mathcal{Y}$ $\quad\quad$ $\hat{f}(\mathbf{X})$

Data D

$$
\begin{gathered}
(\mathbf{X}_1, \mathrm{Y}_1) \\
(\mathbf{X}_2, \mathrm{Y}_2) \\
\vdots \\
(\mathbf{X}_n, \mathrm{Y}_n)
\end{gathered}
$$

- Obtain/train a **classifier** $\hat{f}$
  - without knowing $P_{\mathbf{X}\mathrm{Y}}$, but
  - with **data** D consisting of *i.i.d.* samples of $(\mathbf{X}, \mathrm{Y})$ and *independent* of $(\mathbf{X}, \mathrm{Y})$.
- Fundamental questions:
  - What is a good classifier?
  - How to train a good classifier?

Estimate the probability of head

Unknown
$p := P[\mathrm{Head}]$

Toss n times $\longrightarrow$ H, T, H, … $\xrightarrow{\text{Estimate } p}$ $\hat{\mathrm{p}}$

- From the outcomes of n independent coin tosses, how to estimate the probability p of the coin coming up head?

  $\hat{\mathrm{p}} :=$____ in terms of the number N_H of heads in n coin tosses.

- How good is the estimate?

Estimate expectation by sample average

- Given i.i.d. n-sample of Z
$$\mathrm{Z}^n \coloneqq (\mathrm{Z}_i | i \in [n]) = (\mathrm{Z}_1, \dots, \mathrm{Z}_n),$$
estimate the expectation $E[Z]$ by
$$\bar{\mathrm{Z}} \coloneqq \frac{1}{n} \sum_{i=1}^{n} \mathrm{Z}_i \,, \qquad \text{(sample average)}$$

- The estimate is
  - Unbiased, i.e.,
$$E[\bar{\mathrm{Z}}] = E[\mathrm{Z}]$$
  - Consistent, i.e.,
$$\lim_{n \to \infty} \bar{\mathrm{Z}} = E[\mathrm{Z}]$$

- $\hat{\mathrm{p}} \coloneqq \frac{1}{n} \sum_{i=1}^{n} \mathrm{Z}_i$ and $E[\mathrm{Z}] = p$ by defining the indicator random variable
$$\mathrm{Z} \coloneqq \begin{cases} 1, & \text{—} \\ 0, & \text{—} \end{cases}$$

Empirical Risk Minimization (ERM)

- Estimate risk from data
  - **Empirical risk**:

$$\widehat{\mathrm{R}}(w) := \frac{1}{|\mathrm{D}|} \sum_{(\boldsymbol{x}, y) \in \mathrm{D}} L\big(y, f_w(\boldsymbol{x})\big) = \frac{1}{n} \sum_{i=1}^{n} L(\mathrm{Y}_i, f_w(\mathbf{X}_i))$$

  - **Empirical error probability** with 0-1 loss:

$$\widehat{\mathrm{R}}(w) = \frac{1}{|\mathrm{D}|} \sum_{(\boldsymbol{x}, y) \in \mathrm{D}} \mathbb{1}\big(y \neq f_w(\boldsymbol{x})\big) = \frac{|\{i \in [n] | \mathrm{Y}_i \neq f_w(\mathbf{X}_i)\}|}{n}$$

- Choose the best classifier $f_{\widehat{\mathrm{w}}}$ where

$$\widehat{\mathrm{w}} \in \arg\min_{w \in \mathcal{W}} \widehat{\mathrm{R}}\,(w)$$

- How good is $\hat{\mathrm{f}}$?______________________________

Performance estimate

$$\widehat{\mathrm{R}}(\widehat{\mathrm{w}}) \text{ is a good estimate of } E\left[L\left(\mathrm{Y}, f_{\widehat{\mathrm{w}}}(\mathbf{X})\right)\right]?$$

- Is the empirical risk consistent, i.e.,

$$\widehat{\mathrm{R}}(\widehat{\mathrm{w}}) = \frac{1}{n}\sum_{i=1}^{n} L(\mathrm{Y}_i, f_{\widehat{\mathrm{w}}}(\mathbf{X}_i)) \xrightarrow{n\to\infty} E\left[L\left(\mathrm{Y}, f_{\widehat{\mathrm{w}}}(\mathbf{X})\right)\right]?$$

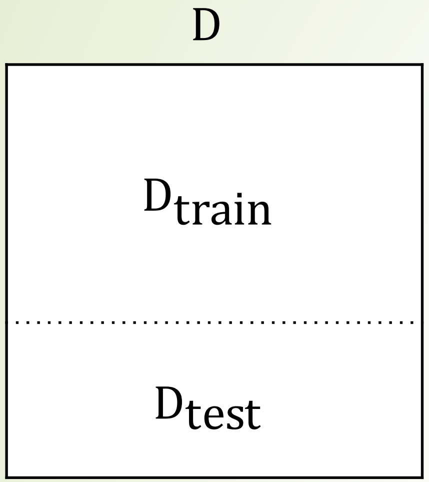
______ because $L\left(\mathrm{Y}_i, f_{\widehat{\mathrm{w}}}(\mathbf{X}_i)\right)$ ______ independent over $i \in [n]$.

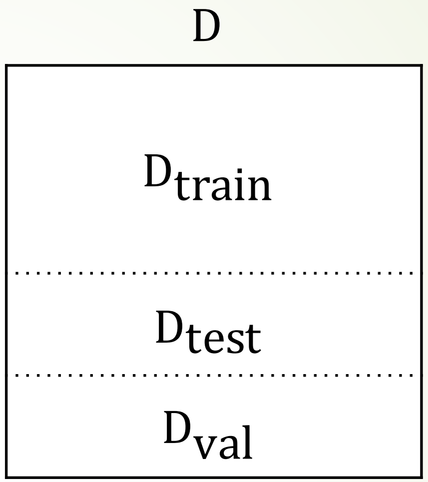
- Is the empirical risk unbiased, i.e.,

$$\mathrm{E}[\widehat{\mathrm{R}}(\widehat{\mathrm{w}})] = \mathrm{E}\left[\frac{1}{n}\sum_{i=1}^{n} L(\mathrm{Y}_i, f_{\widehat{\mathrm{w}}}(\mathbf{X}_i))\right] = \frac{1}{n}\sum_{i=1}^{n} E[L(\mathrm{Y}_i, f_{\widehat{\mathrm{w}}}(\mathbf{X}_i))] = E\left[L\left(\mathrm{Y}, f_{\widehat{\mathrm{w}}}(\mathbf{X})\right)\right]?$$

______ because $L\left(\mathrm{Y}_i, f_{\widehat{\mathrm{w}}}(\mathbf{X}_\mathrm{i})\right)$ ______ identically distributed as $E\left[L\left(\mathrm{Y}, f_{\widehat{\mathrm{w}}}(\mathbf{X})\right)\right]$.

Holdout method





- Hold out some data for testing.
  - D_{train} : **Training set** for constructing the classifier.
  - D_{test} : **Test set** for testing the classifier.
  - D_{val}: **Validation set** sometimes for model selection.
- Usually $2{:}1$ or $2{:}1{:}1$ split. With abundant data, can be $9{:}1$.

Empirical risk on a separate test set

- Train a classifier on $\mathrm{D}_{\mathrm{train}}$, e.g.,
$$\hat{\mathrm{w}} \in \arg\min_{f:\mathcal{X}\to\mathcal{Y}} \frac{1}{|\mathrm{D}_{\mathrm{train}}|} \sum_{(\boldsymbol{x},y)\in \mathrm{D}_{\mathrm{train}}} L(y, f(\boldsymbol{x})).$$
- Compute the empirical risk on $\mathrm{D}_{\mathrm{test}} := \left(\left(\mathbf{X}_{s_i}, \mathrm{Y}_{s_i}\right) | i \in [n'] \right)$
$$\hat{\mathrm{R}}(w) := \frac{1}{|\mathrm{D}_{\mathrm{test}}|} \sum_{(\boldsymbol{x},y)\in \mathrm{D}_{\mathrm{test}}} \mathbb{1}(y \neq f(\boldsymbol{x})) = \frac{1}{n'} \sum_{i=1}^{n} L\left(\mathrm{Y}_{s_i}, f_w\left(\mathbf{X}_{s_i}\right)\right)$$
  - which is unbiased, i.e.,
  $$\mathrm{E}[\hat{\mathrm{R}}(\hat{\mathrm{w}})] = \mathrm{E}\left[\frac{1}{n'} \sum_{i=1}^{n} L\left(\mathrm{Y}_{s_i}, f_{\hat{\mathrm{w}}}\left(\mathbf{X}_{s_i}\right)\right)\right] = \frac{1}{n'} \sum_{i=1}^{n} E\left[L\left(\mathrm{Y}_{s_i}, f_{\hat{\mathrm{w}}}\left(\mathbf{X}_{s_i}\right)\right)\right] = E[L(\mathrm{Y}, f_{\hat{\mathrm{w}}}(\mathbf{X}))]$$
  because $\hat{\mathrm{w}}$ is __________ of $\hat{\mathrm{R}}$ as $\mathrm{D}_{\mathrm{train}}$ and $\mathrm{D}_{\mathrm{test}}$ are independent.
  - and consistent, i.e.,
  $$\hat{\mathrm{R}}(\hat{\mathrm{w}}) = \frac{1}{n'} \sum_{i=1}^{n} L\left(\mathrm{Y}_{s_i}, f_{\hat{\mathrm{w}}}\left(\mathbf{X}_{s_i}\right)\right) \xrightarrow{n' \to \infty} E\left[L\left(\mathrm{Y}_{s_i}, f_{\hat{\mathrm{w}}}\left(\mathbf{X}_{s_i}\right)\right)\right]$$
  by the uniform law of large number if $f_w(\boldsymbol{x})$ is satisfies some conditions. (See Theorem 2 here)

Fails on new values

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1

Training set

	X_1	X_2	Y	$\hat{Y}$	err.
2.	1	26	0	0	
4.	1	27	1	1	

Error rate= _______

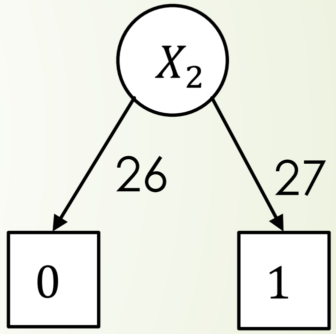
Accuracy= _______%

Test set

	X_1	X_2	Y	$\hat{Y}$	err.
1.	0	25	0	?	+
3.	1	22	1	?	+

Accuracy= ______%

Classifier



Splitting point for numeric attribute

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1

Training set

	X_1	X_2	Y	$\hat{Y}$	err.
2.	1	26	0	0	
4.	1	27	1	1	

Error rate $= \frac{0}{2}$

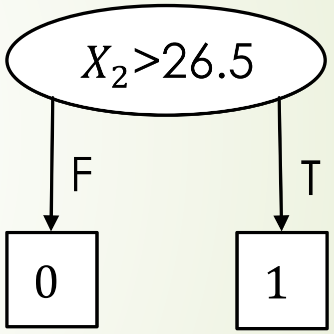
Accuracy= 100%

Test set

	X_1	X_2	Y	$\hat{Y}$	err.
1.	0	25	0	0	
3.	1	22	1	0	+

Accuracy= ______%

Classifier



Is 50% accuracy good/bad?

- $Y = 0$ 50% of the time.
- A classifier making **a**________________ **choices** gives 50% accuracy.

A different split

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1

Training set

	X_1	X_2	Y	$\hat{Y}$	err.
1.	0	25	0	0	
2.	1	26	0	0	

Error rate = ________

Accuracy= ________%

Classifier

$\boxed{0}$

M______________

(**ZeroR** in WEKA)

Test set

	X_1	X_2	Y	$\hat{Y}$	err.
3.	1	22	1	0	+
4.	1	27	1	0	+

Accuracy= ________%

- Different splits likely give different classifiers and accuracies.

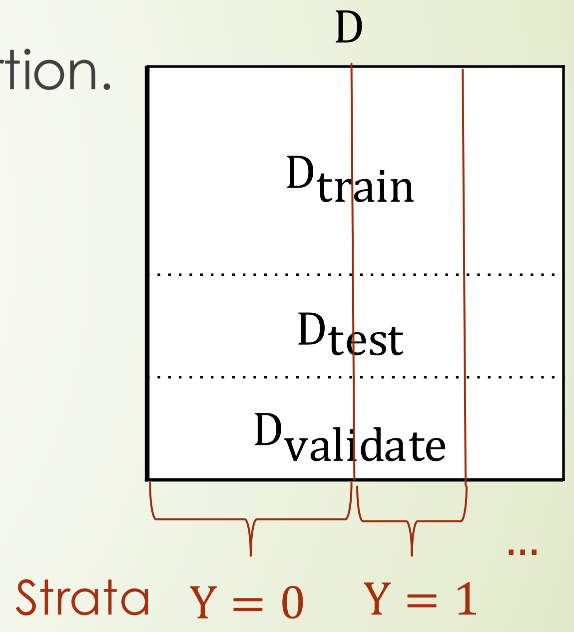
Class proportion

If the class proportion is not maintained,

- test set does not reflect the actual performance because $L\big(y, f_{\widehat{\mathrm{w}}}(\boldsymbol{x})\big)$ for $(\boldsymbol{x}, y) \in \mathrm{D}_{\mathrm{test}}$ may not be ____________________ as $L\big(\mathrm{Y}, f_{\widehat{\mathrm{w}}}(\mathbf{X})\big)$.
- training/validation set can also mislead the learning process.

Stratification

- Sample different classes (strata) independently so samples from different classes maintain the class proportion.
- E.g., **Stratified holdout** maintain the proportions of class values in training/test/validation sets.



How to estimate the typical performance?

- Obtain $N > 1$ random splits and average the performance.
- **Random s**_________________:
  - Randomly sampling **without replacement** for training and test sets.
- Training/test set may still be too small.

Can we make training/test sets larger?

- **B________________**:
  - “pull oneself up by one’s bootstraps”
  - Sampling **with replacement** for training set with the same size as D.
  - Remaining **unsampled data** for test set.

63.2 bootstrap

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1

Training set

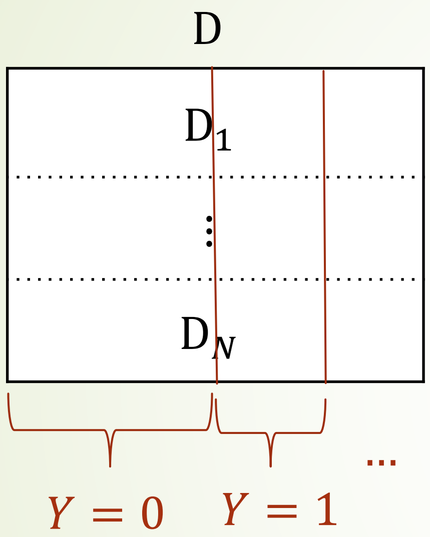
	X_1	X_2	Y
2.	1	26	0
4.	1	27	1
4.	1	27	1
1.	0	25	0

Test set

	X_1	X_2	Y
3.	1	22	1

- Bootstrap n training samples.
- Expected portion of data for test is ______________________________.

N-fold stratified cross-validation



Test: D_1, Training: $D_2, \dots, D_N$
⋮
Test: D_j, Training: $D_1, \dots D_{j-1}, D_{j+1}, \dots, D_N$
⋮
Test: D_N, Training: $D_1, \dots, D_{N-1}$

1. Split data into N parts/folds equally maintaining the class proportion.
2. Repeatedly take a different fold to test on a model trained on all other parts.
3. Report the average performance on the N folds.

Example of 2-fold stratified cross-validation

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1

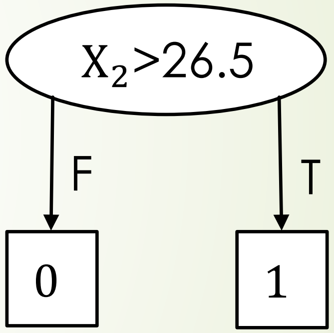
Training set

	X_1	X_2	Y	$\hat{Y}$	err.
2.	1	26	0	0	
4.	1	27	1	1	

Error rate= $\frac{0}{2}$

Accuracy= 100%

Classifier



Test set

	X_1	X_2	Y	$\hat{Y}$	err.
1.	0	25	0	0	
3.	1	22	1	0	+

Accuracy= $\frac{1}{2} = 50\% =: a_1$

Example of 2-fold stratified cross-validation

	X_1	X_2	Y
1.	0	25	0
2.	1	26	0
3.	1	22	1
4.	1	27	1

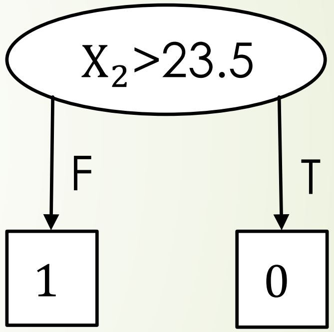
Training set

	X_1	X_2	Y	$\hat{Y}$	err.
1.	0	25	0	0	
3.	1	22	1	1	

Error rate= $\frac{0}{2} = 0\%$

Accuracy= 100%

Classifier



Test set

	X_1	X_2	Y	$\hat{Y}$	err.
2.	1	26	0	0	
4.	1	27	1	0	+

Accuracy= $\frac{1}{2} = 50\% =: a_2 = a_1 = \bar{a}$

Deployment

- Deploy the classifier if performance is good enough, else improve the training.
- **How to deploy?**
  - N-tests give N classifiers. Which one to deploy?
  - Construct a final classifier using the **e**___________ data set, i.e., $f_{\hat{\mathrm{w}}}$ where
$$\hat{\mathrm{w}} \in \arg\min_{\mathrm{w} \in \mathcal{W}} \frac{1}{|\mathrm{D}|} \sum_{(\boldsymbol{x}, y) \in \mathrm{D}} L\big(y, f_{w}(\boldsymbol{x})\big)$$
- **How to improve the training?**
  - Consider different learning algorithms and parameters tuning.
  - Improve the pre-processing of the data.

Controversies?

- Wouldn't the deployed classifier trained on the entire data set suffer from overfitting?
- What about deploying $f_{\hat{\mathrm{w}}}$ where

$$\hat{\mathrm{w}} \in \arg\min_{\mathrm{w} \in \mathcal{W}} \frac{1}{|\mathrm{D}_{\mathrm{train}}|} \sum_{(\boldsymbol{x}, y) \in \mathrm{D}_{\mathrm{train}}} L\big(y, f_{\mathrm{w}}(\boldsymbol{x})\big)\ ?$$

  - Does cross validation prevent overfitting?
  - Why different learning algorithms matter?

References

- 8.5 Model Evaluation and Selection
- Optional reading:
  - Vapnik, Vladimir. "Principles of risk minimization for learning theory." *Advances in neural information processing systems*. 1992.
  - Surrogate loss functions: https://en.wikipedia.org/wiki/Loss_functions_for_classification