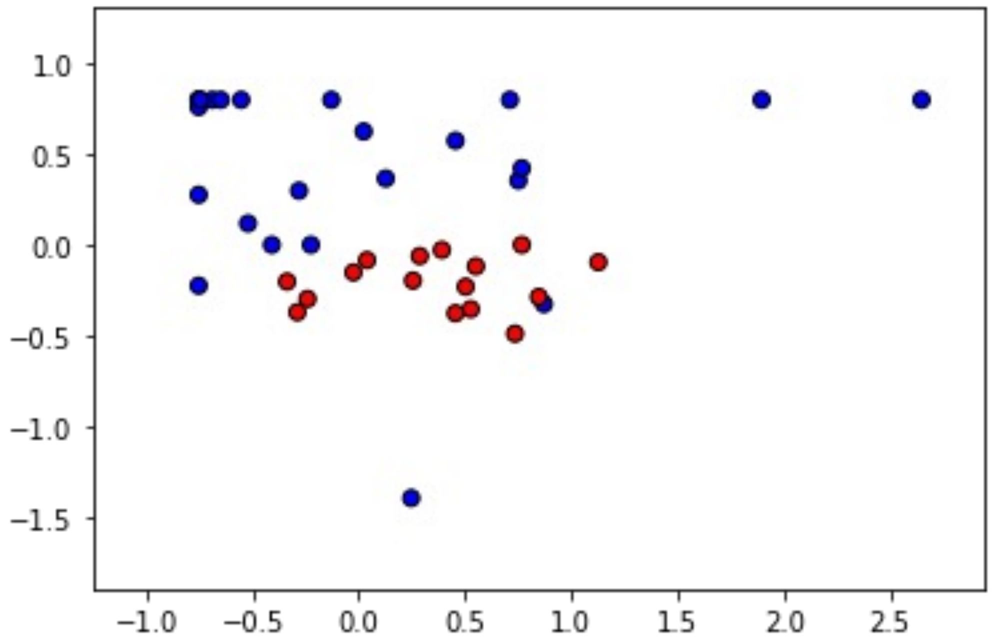
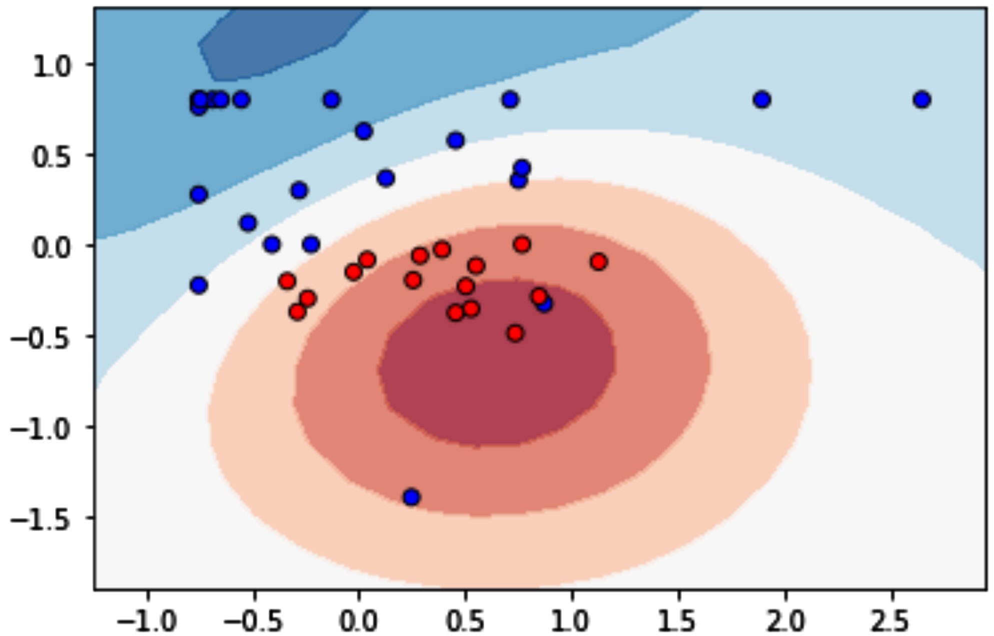
Overfitting and Cross Validation

Some slides borrowed from Nils Hammerla

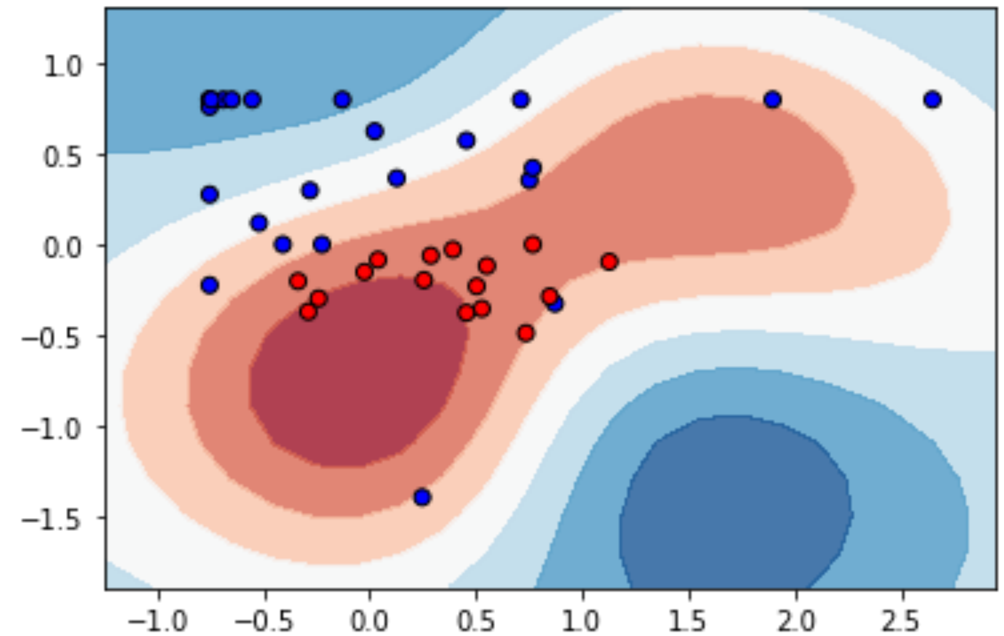
Overfitting



Overfitting

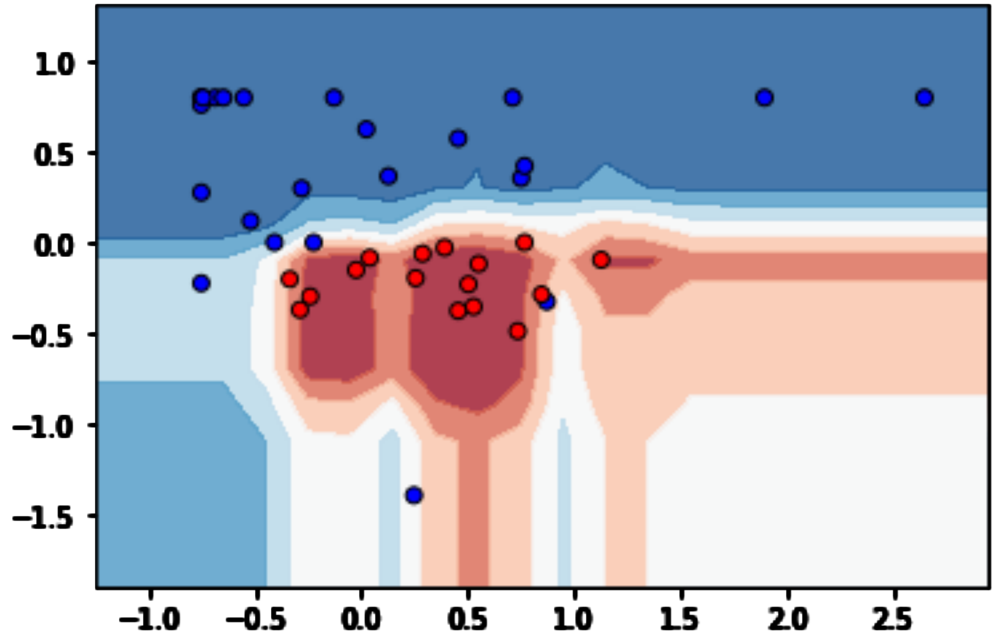


SVM with low confidence



SVM with high confidence

Overfitting

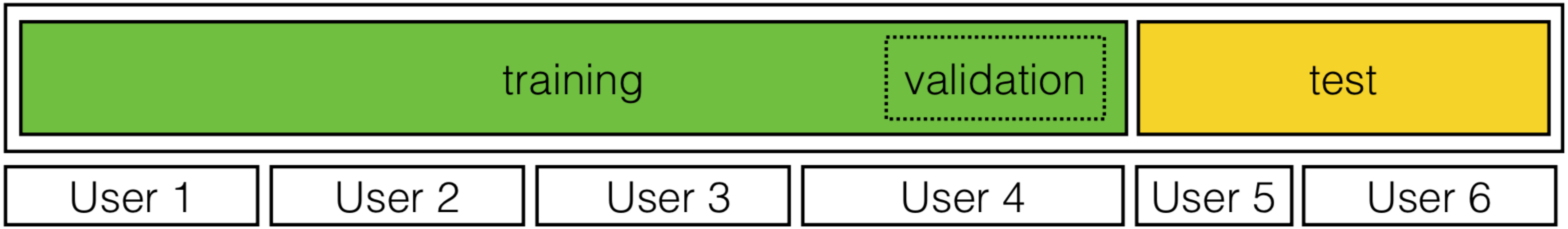


Training and test-sets

Training and test-sets

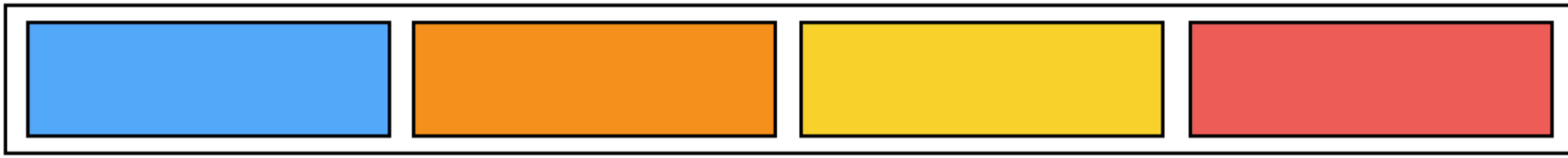
- **Hold-out validation**
  - Choose part of the data as training and test-set
    - By some heuristic (e.g. 20% test),
    - Depending on study design.
  - Gives a single performance estimate
  - Performance may critically depend on chosen test-set

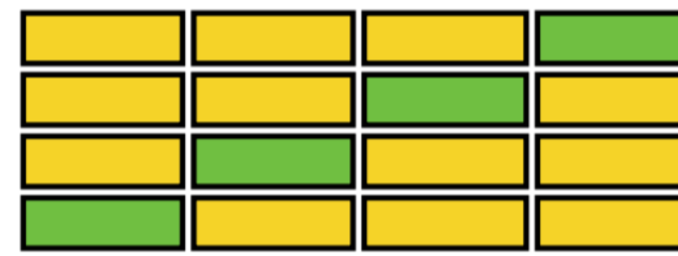
Training and test-sets



- **Hold-out validation**
  - Choose part of the data as training and test-set
    - By some heuristic (e.g. 20% test),
    - Depending on study design.
  - Gives a single performance estimate
  - Performance may critically depend on chosen test-set

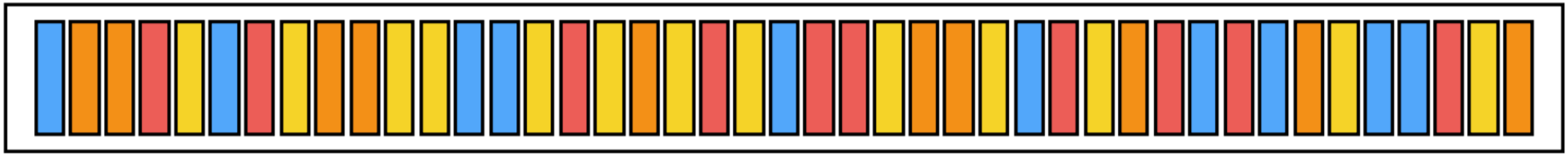
Training and test-sets





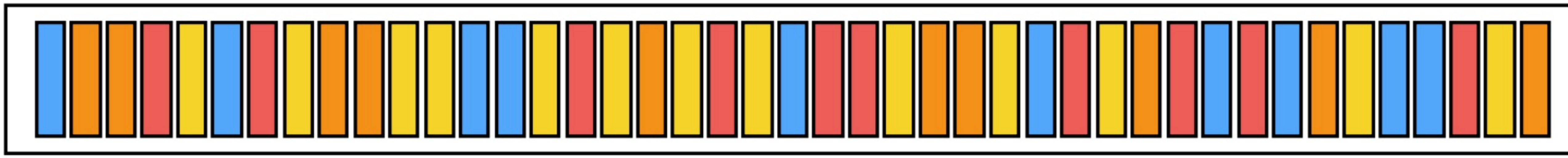
- **Repeated hold-out**
  - Split data into k continuous *folds*
  - A split into n folds leads to n performance estimates
  - Gives more reliable performance estimate
    - But: this depends on how the folds are constructed!
  - Variants
    - Leave-one-subject-out (LOSO) — User-independent
    - Leave-one-run-out — User-dependent

Training and test-sets

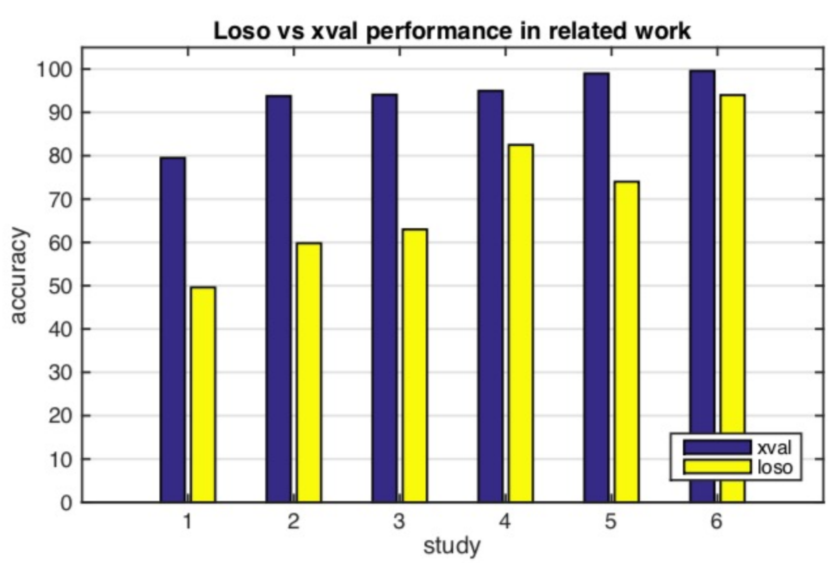


- **(Random, stratified) cross-validation**
  - Split data into k folds, on the lowest level
  - Folds constructed to be *stratified* w.r.t. class distribution
  - Popular in general machine learning
  - Standard approach in many ML frameworks
  - User-dependent performance estimate

Pitfalls



- **User dependent vs independent performance**
  - Assumption: User dependent performance is the upper bound of possible system performance
  - Cross-validation therefore popular in ubicomp to demonstrate feasibility of e.g. new technical approach

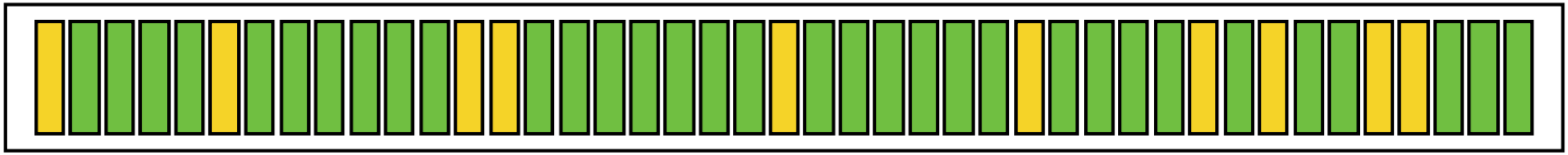


Cross-Validation Test

We tested the performance of the algorithm by 10-fold cross validation including all participants from the first experiment. By using DFT and gravity tilt features together, we were able to obtain near-perfect overall accuracy of 99.6% in cross-validation. When only the DFT features were used,

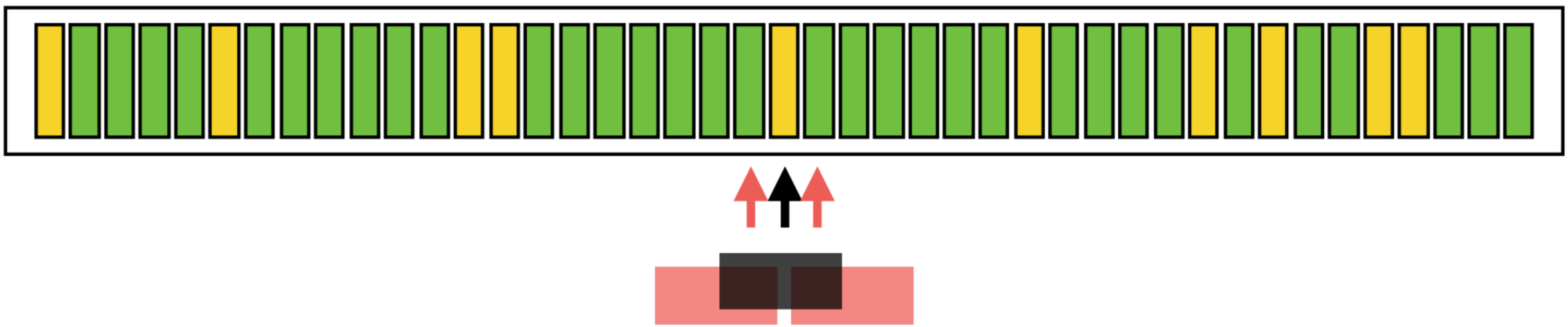
The results show 93.8% [redacted] activity recognition accuracy when using a person-dependent classifier and 59.8% accuracy when using a person-independent classifier.

Pitfalls



What happens in case of time series data?

Pitfalls



- **Cross-validation in segmented time-series**
  - When testing for a segment *i*, it is very likely that segments *i-1* or *i +1* are in the training set.
  - Neighbouring segments typically overlap
    - They are therefore very similar!
  - This biases the results and leads to bloated performance figures

Data Collection Sessions

- Sometimes you need to collect same data multiple times, in different sessions.
- Example:

4	1	20	3
Appliances	Minute	Instances	Sessions

Performance metrics

prediction

ground truth	A	B	C
A	1022	13	144
B	300	542	24
C	12	55	132

Basic elements for each class:

- true positives
- false positives
- false negatives
- true negatives

Performance metrics

For each class (or for two class problems):

Precision / PPV	tp / (tp + fp)
Recall / Sensitivity	tp / (tp + fn)
Specificity	tn / (tn + fp)
Accuracy	(tp+tn) / (tp + fp + fn + tn)
F1-score	2*prec*sens / (prec*sens)

Beyond Accuracy

- Accuracy = $\dfrac{\text{How many times am I } \textbf{correct}\text{?}}{\text{Total number of times}}$

But, correct about what? There are two decisions here.

- Precision and Recall suggest that we look at performance from the perspective of positive inferences

Recall is
what proportion of actual positive cases were correctly identified?

$$\frac{\text{correctly identified positive cases}}{\text{actual positive cases}}$$

Performance metrics

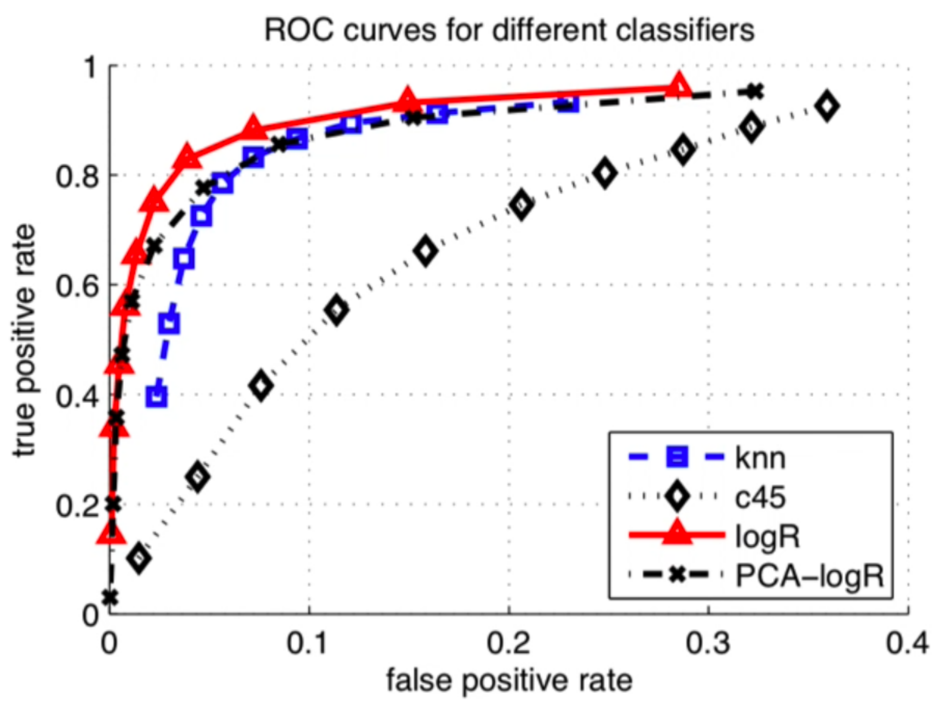
For a set of classes C = {A,B,C,…}:

Overall accuracy	$\frac{1}{N}\sum_{c\in C} tp_c$
Mean accuracy	$\frac{1}{\lvert C\rvert}\sum_{c\in C} \mathrm{acc}_c$
Weighted F1-score	$\frac{2}{\lvert C\rvert}\sum_{c\in C} \frac{\mathrm{prec}_c \times \mathrm{sens}_c}{\mathrm{prec}_c + \mathrm{sens}_c}$
Mean F1-score	$\frac{2}{N}\sum_{c\in C} n_c \frac{\mathrm{prec}_c \times \mathrm{sens}_c}{\mathrm{prec}_c + \mathrm{sens}_c}$

ROC Curves

Receiver Operator Characteristics (ROC)

- Illustrates **trade-off** between **True Positive Rate** (sensitivity / recall), and **False Positive Rate** (1 - specificity)
- Useful if approach has a simple parameter, like a threshold.



[Ladha, Cassim, et al. "ClimbAX: skill assessment for climbing enthusiasts." Ubicomp 2013]

Bland Altman Plots