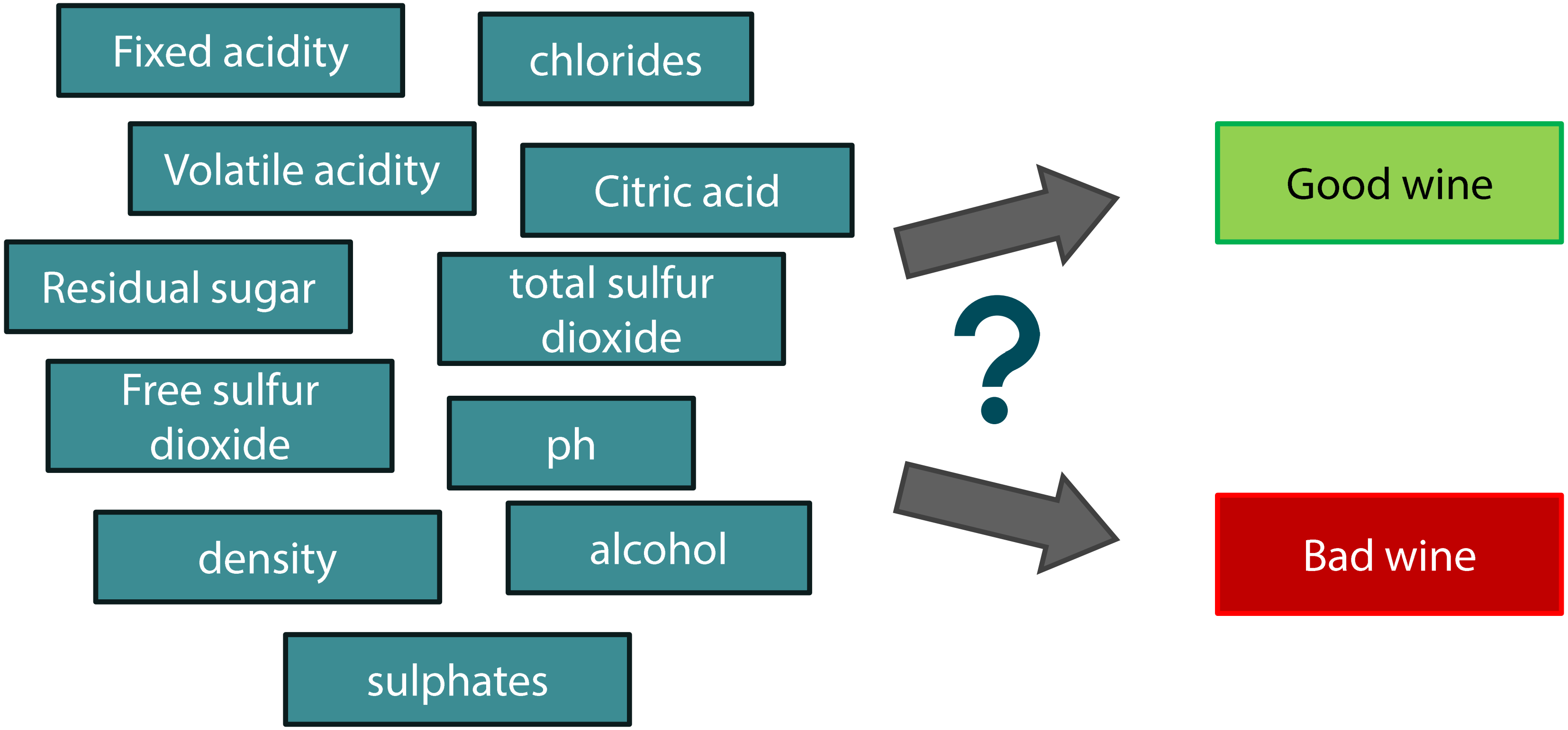
UNIVERSITÄT ZU LÜBECK

Institut für Robotik und Kognitive Systeme | Daniel Wulff

Artificial Intelligence II

Exercise Sheet 9

Wine classification



Fixed acidity

chlorides

Volatile acidity

Citric acid

Residual sugar

total sulfur dioxide

Free sulfur dioxide

ph

density

alcohol

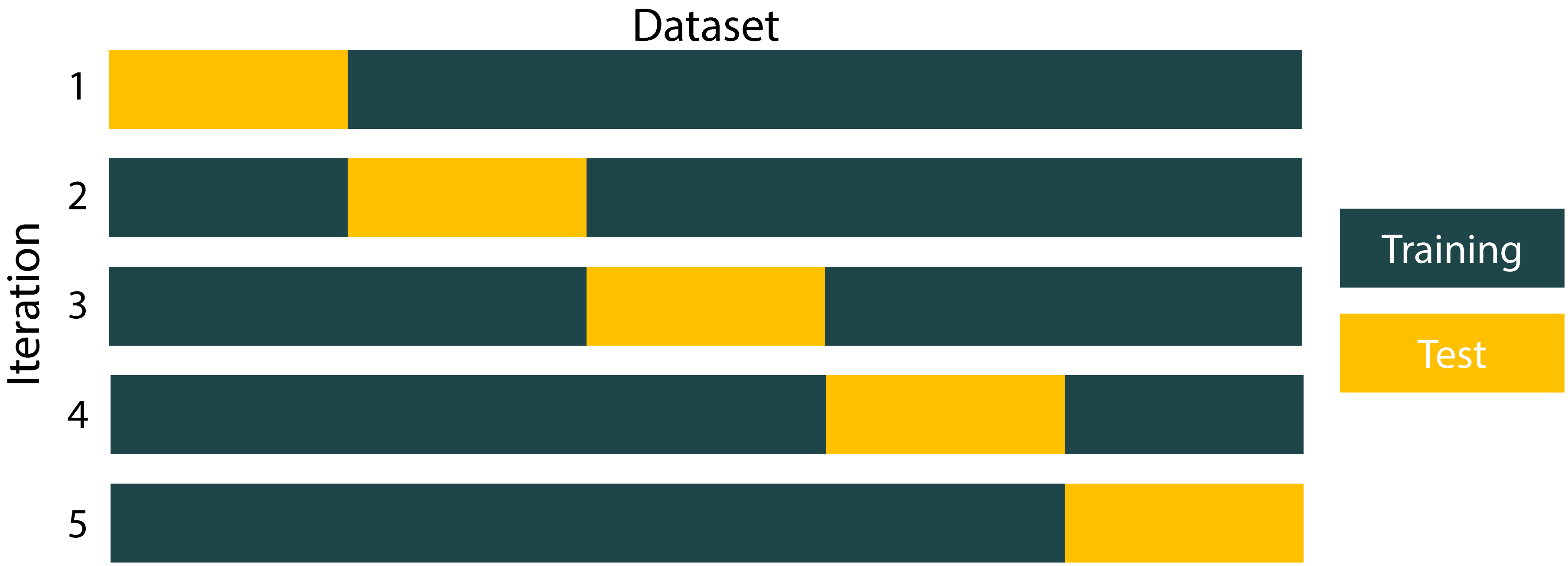
sulphates

?

Good wine

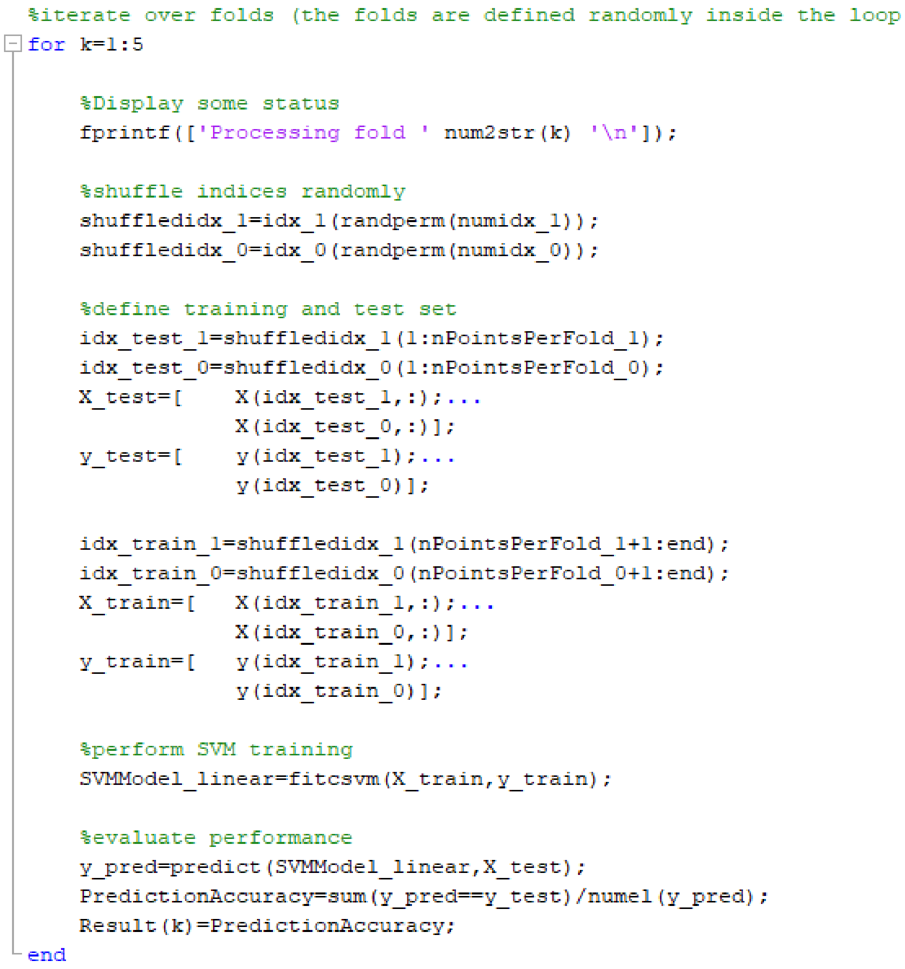
Bad wine

5-fold Cross validation



- Validate generalizability
- Training and testing in 5 different settings

Linear Classification



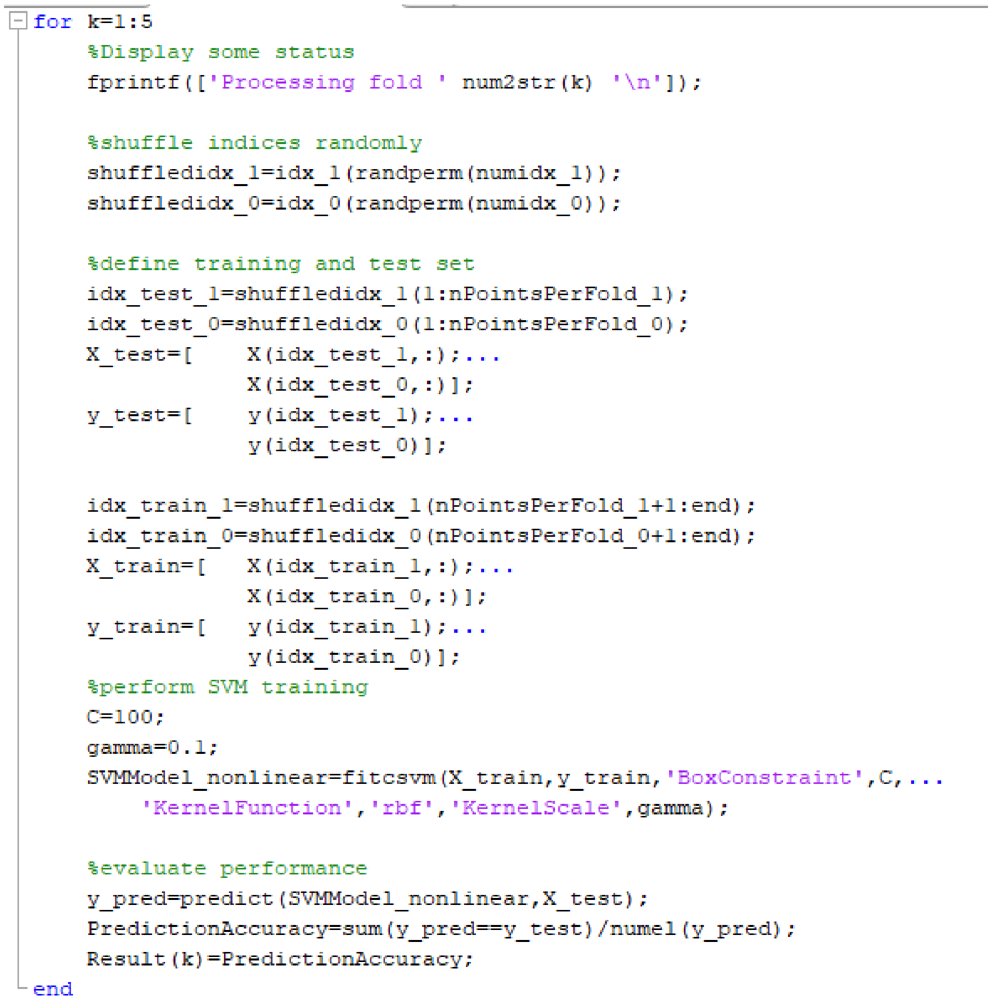
```
%iterate over folds (the folds are defined randomly inside the loop
for k=1:5

    %Display some status
    fprintf(['Processing fold ' num2str(k) '\n']);

    %shuffle indices randomly
    shuffledidx_1=idx_1(randperm(numidx_1));
    shuffledidx_0=idx_0(randperm(numidx_0));

    %define training and test set
    idx_test_1=shuffledidx_1(1:nPointsPerFold_1);
    idx_test_0=shuffledidx_0(1:nPointsPerFold_0);
    X_test=[    X(idx_test_1,:);...
                X(idx_test_0,:)];
    y_test=[    y(idx_test_1);...
                y(idx_test_0)];

    idx_train_1=shuffledidx_1(nPointsPerFold_1+1:end);
    idx_train_0=shuffledidx_0(nPointsPerFold_0+1:end);
    X_train=[   X(idx_train_1,:);...
                X(idx_train_0,:)];
    y_train=[   y(idx_train_1);...
                y(idx_train_0)];

    %perform SVM training
    SVMModel_linear=fitcsvm(X_train,y_train);

    %evaluate performance
    y_pred=predict(SVMModel_linear,X_test);
    PredictionAccuracy=sum(y_pred==y_test)/numel(y_pred);
    Result(k)=PredictionAccuracy;
end
```

Performance:

1. 0.87
2. 0.87
3. 0.87
4. 0.87
5. 0.87

Mean: 0.87

Nonlinear Classification



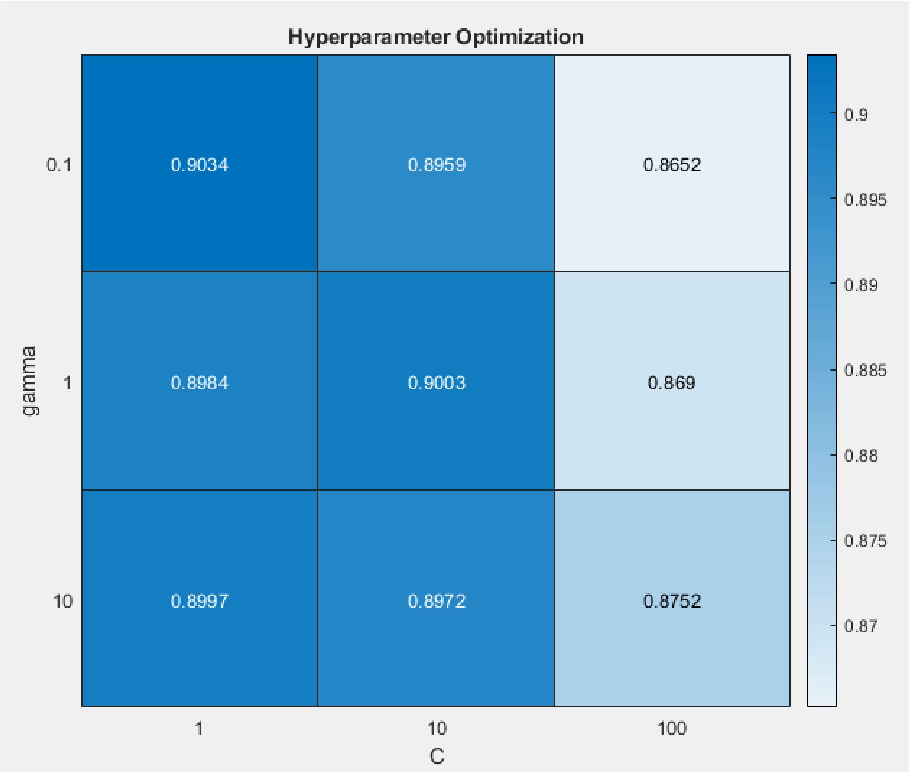
```
for k=1:5
    %Display some status
    fprintf(['Processing fold ' num2str(k) '\n']);

    %shuffle indices randomly
    shuffledidx_1=idx_1(randperm(numidx_1));
    shuffledidx_0=idx_0(randperm(numidx_0));

    %define training and test set
    idx_test_1=shuffledidx_1(1:nPointsPerFold_1);
    idx_test_0=shuffledidx_0(1:nPointsPerFold_0);
    X_test=[    X(idx_test_1,:);...
                X(idx_test_0,:)];
    y_test=[    y(idx_test_1);...
                y(idx_test_0)];

    idx_train_1=shuffledidx_1(nPointsPerFold_1+1:end);
    idx_train_0=shuffledidx_0(nPointsPerFold_0+1:end);
    X_train=[   X(idx_train_1,:);...
                X(idx_train_0,:)];
    y_train=[   y(idx_train_1);...
                y(idx_train_0)];
    %perform SVM training
    C=100;
    gamma=0.1;
    SVMModel_nonlinear=fitcsvm(X_train,y_train,'BoxConstraint',C,...
        'KernelFunction','rbf','KernelScale',gamma);

    %evaluate performance
    y_pred=predict(SVMModel_nonlinear,X_test);
    PredictionAccuracy=sum(y_pred==y_test)/numel(y_pred);
    Result(k)=PredictionAccuracy;
end
```

Performance:
1. 0.90
2. 0.91
3. 0.91
4. 0.90
5. 0.90

Mean: 0.90

$C = 1, \gamma = 0.01$

Hyperparameter Tuning



Next Lectures

- January 29: No lecture
- February 5: Repetitorium
  - Send me your questions till February 3!!
    Email: wulff@rob.uni-luebeck.de