Module 05: Supervised Learning Recap — Cheat Sheet

ML for Innovation Research — Prof. J. Osterrieder

Supervised vs. Unsupervised

	Supervised	Unsupervised
Labels	Required	Not needed
Goal	Predict y from X	Discover structure
Output	Prediction / class	Clusters / topics
Evaluation	Accuracy, F1, AUC	Silhouette, coherence
Risk	Overfitting	Subjective interpretation
Examples	Random Forest, Logistic Reg.	K-Means, PCA, LDA

Features vs. Labels

Features (X)
Input variables the model uses to make predictions.
Can be numerical, categorical, or text-derived.

Label (y)
The target variable you want to predict.
Classification: discrete classes (e.g. churn / no-churn).
Regression: continuous value (e.g. revenue).

Overfitting

Definition: The model learns the training data too well, including its noise, and fails on unseen data.

Symptoms:

- High training accuracy, low test accuracy.
- Large gap between train and validation performance.
- Model complexity far exceeds data size.

Prevention:

- Use train/test split or cross-validation.
- Regularisation (L1/L2 penalties).
- Simpler models (fewer parameters).
- More training data.
- Feature selection (remove irrelevant features).

Train/Test Split

- Hold out 20–30% of data for testing.
- **Never** evaluate on training data.
- Stratify when classes are imbalanced.

```
from sklearn.model_selection import \
    train_test_split

X_train, X_test, y_train, y_test = \
    train_test_split(X, y, test_size=0.2,
                     stratify=y,
                     random_state=42)
```

Cross-Validation

- Split data into k folds (typically k=5).
- Train on $k-1$ folds, validate on the remaining fold.
- Repeat k times; report mean $\pm$ std.
- More robust estimate than a single split.

```
from sklearn.model_selection import \
    cross_val_score

scores = cross_val_score(model, X, y,
                         cv=5,
                         scoring='accuracy')
print(f"{scores.mean():.3f} +/- "
      f"{scores.std():.3f}")
```

Split vs. Cross-Validation

Train/Test Split	Cross-Validation
Fast, one evaluation	k evaluations
High variance estimate	More stable estimate
Good for large datasets	Better for small datasets
Use for final reporting	Use for model selection

Golden Rule: Never let test data influence training decisions (feature engineering, hyperparameter tuning, or model selection).