Test and Score - Orange

Cross validation
- Number of folds: 5
- ☑ Stratified
- ○ Cross validation by feature
- ○ Random sampling
  - Repeat train/test: 10
  - Training set size: 66 %
  - ☑ Stratified
- ○ Leave one out
- ○ Test on train data
- ○ Test on test data

Evaluation results for target: (None, show average over classes)

Model	AUC	CA	F1	Prec	Recall	MCC
Constant	0.494	0.259	0.106	0.067	0.259	0.000
kNN	0.981	0.875	0.875	0.884	0.875	0.844
SVM	0.994	0.918	0.919	0.921	0.918	0.897
Random Forest	0.971	0.873	0.872	0.876	0.873	0.839
Logistic Regression	0.995	0.932	0.932	0.932	0.932	0.914
Neural Network	0.994	0.932	0.932	0.932	0.932	0.914
CN2 Rule Induction	0.873	0.712	0.710	0.714	0.712	0.634
Gradient Boosting	0.984	0.905	0.904	0.904	0.905	0.879

Compare models by: Area under ROC curve   ☐ Negligible diff.: 0.1

	Constant	kNN	SVM	Random Forest	Logistic Regr...	Neural Netw...	CN2 Rule Ind...	Gradient Boo...
Constant		0.000	0.000	0.000	0.000	0.000	0.000	0.000
kNN	1.000		0.027	0.864	0.016	0.019	0.999	0.144
SVM	1.000	0.973		0.995	0.420	0.504	1.000	0.979
Random Forest	1.000	0.136	0.005		0.011	0.008	0.999	0.034
Logistic Regression	1.000	0.984	0.580	0.989		0.708	1.000	0.979
Neural Network	1.000	0.981	0.496	0.992	0.292		1.000	0.982
CN2 Rule Induction	1.000	0.001	0.000	0.001	0.000	0.000		0.000
Gradient Boosting	1.000	0.856	0.021	0.966	0.021	0.018	1.000	

p(Constant > Logistic Regression) = 0.000

Table shows probabilities that the score for the model in the row is higher than that of the model in the column. Small numbers show the probability that the difference is negligible.

441 | - | □□□□□□□□ | - 441 | 8×441

Q: Why linear regression can't use for classification?

A: Linear regression is primarily used for predicting a continuous dependent variable based on one or more independent variables. While it can technically be used for classification tasks, there are several reasons why it's not ideal:

1. Output Range:
   Linear regression predictions can extend beyond the range [0,1], which is not suitable for binary classification tasks where the output is typically a probability between 0 and 1. For classification problems, we are often interested in predicting probabilities. Linear regression does not naturally provide probabilities as it is not bounded between 0 and 1.

2. Sensitive to Outliers:
   Linear regression is sensitive to outliers. In classification tasks, especially those with imbalanced classes, this sensitivity can lead to skewed predictions.
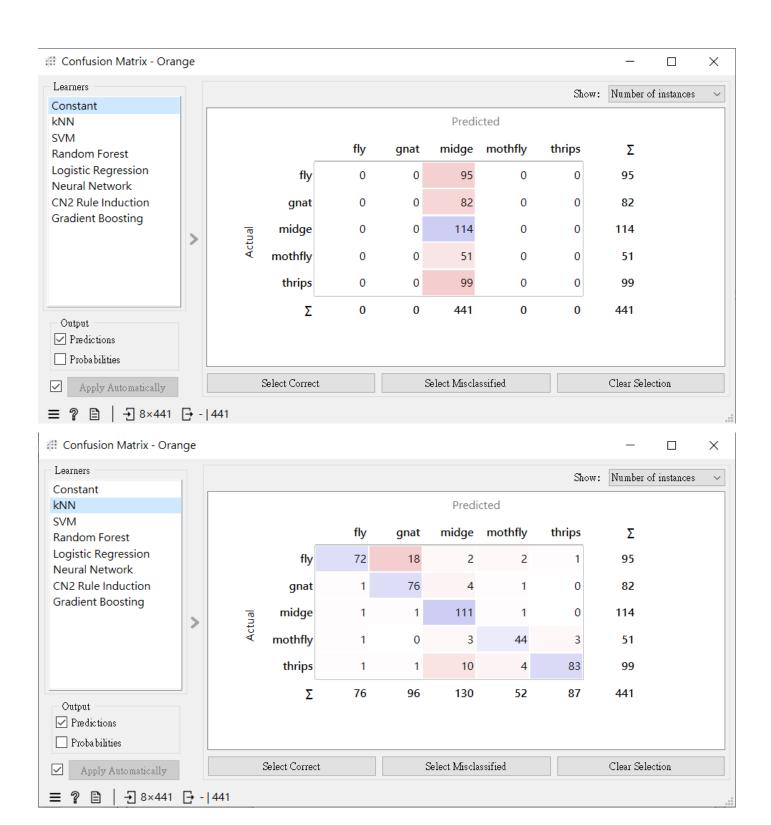
3. Assumption of Linearity:
   Linear regression assumes a linear relationship between the dependent and independent variables. In many classification tasks, this relationship is non-linear, which makes linear regression an ineffective tool.

4. Loss Function:
   The loss function used in linear regression (mean squared error) is not the best choice for classification problems. Classification tasks are better served by loss functions that focus on probabilities and class separations, like cross-entropy.

Other methods like logistic regression, which is specifically designed for classification problems, are preferred. Logistic regression provides outputs between 0 and 1, interprets these outputs as probabilities, and uses a loss function suited for binary outcomes.

Confusion Matrix - Orange

Learners
- Constant
- kNN
- SVM
- Random Forest
- Logistic Regression
- Neural Network
- CN2 Rule Induction
- Gradient Boosting

Show: Number of instances

Predicted

	fly	gnat	midge	mothfly	thrips	Σ
fly	0	0	95	0	0	95
gnat	0	0	82	0	0	82
midge	0	0	114	0	0	114
mothfly	0	0	51	0	0	51
thrips	0	0	99	0	0	99
Σ	0	0	441	0	0	441

Actual

Output
- ☑ Predictions
- ☐ Probabilities
- ☑ Apply Automatically

Select Correct   Select Misclassified   Clear Selection

≡ ? ▤ | ⇥ 8×441 ⇤ - | 441

Confusion Matrix - Orange

Learners
- Constant
- kNN
- SVM
- Random Forest
- Logistic Regression
- Neural Network
- CN2 Rule Induction
- Gradient Boosting

Show: Number of instances

Predicted

	fly	gnat	midge	mothfly	thrips	Σ
fly	72	18	2	2	1	95
gnat	1	76	4	1	0	82
midge	1	1	111	1	0	114
mothfly	1	0	3	44	3	51
thrips	1	1	10	4	83	99
Σ	76	96	130	52	87	441

Actual

Output
- ☑ Predictions
- ☐ Probabilities
- ☑ Apply Automatically

Select Correct   Select Misclassified   Clear Selection

≡ ? ▤ | ⇥ 8×441 ⇤ - | 441

Confusion Matrix - Orange

Learners
- Constant
- kNN
- **SVM**
- Random Forest
- Logistic Regression
- Neural Network
- CN2 Rule Induction
- Gradient Boosting

Show: Number of instances

Predicted

Actual		fly	gnat	midge	mothfly	thrips	Σ
	fly	90	3	1	1	0	95
	gnat	9	72	1	0	0	82
	midge	2	0	108	1	3	114
	mothfly	3	0	1	43	4	51
	thrips	2	0	2	3	92	99
	Σ	106	75	113	48	99	441

Output
- ☑ Predictions
- ☐ Probabilities

☑ Apply Automatically

| Select Correct | Select Misclassified | Clear Selection |

≡ ? 🗎 | → 8×441 → - | 441

Confusion Matrix - Orange

Learners
- Constant
- kNN
- SVM
- **Random Forest**
- Logistic Regression
- Neural Network
- CN2 Rule Induction
- Gradient Boosting

Show: Number of instances

Predicted

Actual		fly	gnat	midge	mothfly	thrips	Σ
	fly	85	8	2	0	0	95
	gnat	6	72	4	0	0	82
	midge	2	1	103	1	7	114
	mothfly	6	1	5	37	2	51
	thrips	2	0	8	1	88	99
	Σ	101	82	122	39	97	441

Output
- ☑ Predictions
- ☐ Probabilities

☑ Apply Automatically

| Select Correct | Select Misclassified | Clear Selection |

≡ ? 🗎 | → 8×441 → - | 441

Confusion Matrix - Orange (Logistic Regression)

Show: Number of instances

		Predicted					
		fly	gnat	midge	mothfly	thrips	Σ
Actual	fly	87	3	2	2	1	95
	gnat	6	74	2	0	0	82
	midge	2	0	110	1	1	114
	mothfly	2	0	1	43	5	51
	thrips	0	0	0	2	97	99
	Σ	97	77	115	48	104	441

Learners: Constant, kNN, SVM, Random Forest, Logistic Regression, Neural Network, CN2 Rule Induction, Gradient Boosting

Output: ☑ Predictions ☐ Probabilities ☑ Apply Automatically

Select Correct | Select Misclassified | Clear Selection

8×441 - |441

Confusion Matrix - Orange (Neural Network)

Show: Number of instances

		Predicted					
		fly	gnat	midge	mothfly	thrips	Σ
Actual	fly	90	3	1	1	0	95
	gnat	6	74	2	0	0	82
	midge	1	2	109	1	1	114
	mothfly	2	0	1	43	5	51
	thrips	0	0	1	3	95	99
	Σ	99	79	114	48	101	441

Learners: Constant, kNN, SVM, Random Forest, Logistic Regression, Neural Network, CN2 Rule Induction, Gradient Boosting

Output: ☑ Predictions ☐ Probabilities ☑ Apply Automatically

Select Correct | Select Misclassified | Clear Selection

8×441 - |441

Confusion Matrix - Orange (CN2 Rule Induction)

Learners: Constant, kNN, SVM, Random Forest, Logistic Regression, Neural Network, **CN2 Rule Induction**, Gradient Boosting

Show: Number of instances

Predicted

Actual \ Predicted	fly	gnat	midge	mothfly	thrips	Σ
fly	68	20	3	2	2	95
gnat	16	53	9	0	4	82
midge	3	6	96	4	5	114
mothfly	5	2	10	27	7	51
thrips	4	5	14	6	70	99
Σ	96	86	132	39	88	441

Output: ☑ Predictions ☐ Probabilities ☑ Apply Automatically

Select Correct | Select Misclassified | Clear Selection

8×441 ⤷ - | 441

Confusion Matrix - Orange (Gradient Boosting)

Learners: Constant, kNN, SVM, Random Forest, Logistic Regression, Neural Network, CN2 Rule Induction, **Gradient Boosting**

Show: Number of instances

Predicted

Actual \ Predicted	fly	gnat	midge	mothfly	thrips	Σ
fly	86	6	1	2	0	95
gnat	7	70	3	2	0	82
midge	1	1	108	3	1	114
mothfly	3	1	0	41	6	51
thrips	1	0	4	0	94	99
Σ	98	78	116	48	101	441

Output: ☑ Predictions ☐ Probabilities ☑ Apply Automatically

Select Correct | Select Misclassified | Clear Selection

8×441 ⤷ - | 441